Braunschweiger Schriften zur Mechanik Nr. 59-2005

A Stabilized and Coupled Meshfree/Meshbased Method for Fluid-Structure Interaction Problems

von

Thomas-Peter Fries

aus Lübeck

Herausgegeben vom Institut für Angewandte Mechanik
der Technischen Universität Braunschweig

Schriftleiter: Prof. H. Antes
Institut für Angewandte Mechanik
Postfach 3329
38023 Braunschweig

ISBN 3-920395-58-1

Vom Fachbereich Bauingenieurwesen
der Technischen Universität Carolo-Wilhelmina zu Braunschweig
zur Erlangung des Grades eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

From the Faculty of Civil Engineering
at the Technische Universität Carolo-Wilhelmina zu Braunschweig
in Brunswick, Germany,
approved dissertation

Eingereicht am:	10.05.2005
Mündliche Prüfung am:	21.07.2005
Berichterstatter:	Prof. H.G. Matthies, Inst. für Wissenschaftliches Rechnen
	Prof. M. Krafczyk, Inst. für Computeranwendungen im Bauingenieurwesen

Abstract

A method is presented which combines features of meshfree and meshbased methods in order to enable the simulation of complex flow problems involving large deformations of the domain or moving and rotating objects.

Conventional meshbased methods like the finite element method have matured as standard tools for the simulation of fluid and structure problems. They offer efficient and reliable approximations, provided that a conforming mesh with sufficient quality can be maintained throughout the simulation. This, however, may not be guaranteed for complex fluid and fluid-structure interaction problems.

Meshfree methods on the other hand approximate partial differential equations based on a set of nodes without the need for an additional mesh. Therefore, these methods are frequently used for problems where suitable meshes are prohibitively expensive to construct and maintain. This advantage of meshfree methods comes at the price of being considerably more time-consuming than their meshbased counterparts.

A coupled meshfree/meshbased fluid solver is developed which combines the advantages of both methodologies. A meshfree method, closely related to the element-free Galerkin method, is used in small parts of the domain where a mesh is difficult to maintain, whereas the efficient meshbased finite element method is employed in the rest of the domain.

Major steps in the development of the coupled flow solver are the extension of standard stabilization methods to meshfree approximations, and the realization of the coupling on the level of the shape functions, which are involved in the approximation of the governing equations. Concerning the stabilization, it is found that the same structure of the standard stabilization schemes may be used for meshfree approximations, however, the aspect of the stabilization parameter, weighing the stabilization terms, requires special care. For the coupling, these requirements for a reliable stabilization lead to modifications of the existing coupling approaches.

The stabilized and coupled flow solver is verified and used for the simulation of geometrically complex fluid-structure interaction problems. Conventional meshbased methods are not suitable for the approximation of these test cases due to the prohibitively large deformations of the geometry.

Zusammenfassung

Es wird ein Verfahren vorgestellt, das Eigenschaften netzfreier und netzbasierter Methoden nutzt, um die Simulation komplexer Strömungsprobleme zu ermöglichen, bei denen große Verformungen des Gebietes oder sich bewegende und rotierende Objekte in der Strömung berücksichtigt werden können.

Konventionelle netzbasierte Verfahren wie die Finite Element Methode haben sich zu Standardwerkzeugen bei der numerischen Analyse von Strömungen und Festkörpern entwickelt. Sie ermöglichen eine schnelle und zuverlässige Approximation, vorausgesetzt, daß ein konformes Netz mit geeigneter Qualität während der gesamten Simulation aufrecht erhalten werden kann. Dies kann jedoch bei komplexen Strömungs- und Fluid-Struktur-Interaktionsproblemen eine entscheidende Einschränkung in der Anwendbarkeit dieser Verfahren darstellen.

Netzfreie Verfahren approximieren dagegen die zugrundeliegenden Modellgleichungen nur in Abhängigkeit einer gegebenen Knotenverteilung, ohne ein Netz zu erfordern, das die Konnektivität *a priori* festlegt. Deshalb werden diese Verfahren häufig dort eingesetzt, wo die Generierung oder Erhaltung geeigneter Netze nicht mit vertretbarem Aufwand möglich ist. Allerdings geht der Vorteil der Netzunabhängigkeit bei netzfreien Verfahren einher mit deutlich aufwendiger zu konstruierenden Ansatzfunktionen, was eine empfindliche Erhöhung des Rechenaufwandes bei der Integration der zugrundeliegenden Gleichungen erfordert.

Hierin wird ein gekoppelter netzfreier/netzbasierter Strömungslöser entwickelt, der die Vorteile beider Verfahren kombiniert. Ein netzfreies Verfahren, das eng mit der ,,element-free" Galerkin Methode verwandt ist, wird nur in kleinen Teilen des Gebietes verwendet, wo ein Netz zu Schwierigkeiten führt, und im gesamten Rest des Gebiets wird die Finite Element Methode als effizientes netzbasiertes Standardverfahren eingesetzt.

Wichtige Schritte bei der Entwicklung der gekoppelten Methode sind die Erweiterung von Standard-Stabilisierungsansätzen auf netzfreie Verfahren und die Umsetzung der Kopplung auf der Ebene der Ansatzfunktionen, die für die Approximation eingesetzt werden. Bezüglich der Stabilisierung wird gezeigt, daß die Struktur der verschiedenen Stabilisierungsmethoden direkt auf netzfreie Verfahren anwendbar ist, allerdings erfordert die Wahl geeigneter Stabilisierungsparameter, die den Stabilisierungseinfluß wichten, besondere Beachtung. Bei der Kopplung führen die Voraussetzungen, die für eine zuverlässige Stabilisierung gegeben sein müssen, zu Modifikationen der Standardansätze.

Der gekoppelte Strmungslser wird verifiziert und fr die Simulation geometrisch komplexer Fluid-Struktur-Interaktionsprobleme eingesetzt. Die Fhigkeiten des gekoppelten Verfahrens werden dabei sichtbar, denn klassische netzbasierte Standardverfahren versagen bei den gezeigten Anwendungsbeispielen.

Contents

1 Introduction

The motivation for this work is the simulation of complex fluid and fluid-structure interaction problems. A number of methods have been developed for the numerical simulation of flow problems, each with its characteristic advantages and disadvantages. One may characterize these methods based on the fact whether they employ a mesh for the discretization or not. Meshbased methods may be labeled standard tools in the numerical world. Well-known members of this class are the finite volume method (FVM) and the finite element method (FEM). Both these methods are used frequently and reliably for the approximation of the Navier-Stokes equations [46, 70, 82, 191], the governing equations of fluid dynamics. The success of the approximations depends on the ability to construct and maintain meshes of sufficient quality throughout the whole simulation. This, however, may pose serious problems in certain applications, for example in the presence of large geometric deformations of the boundary, or rotating and moving obstacles—situations which may frequently occur in the context of fluid-structure interaction problems. Different elaborate techniques have been developed to deal with these problems in the meshbased context: the fictitious domain [26, 69] and fictitious boundary methods [183], techniques employing overlapping grids [40, 83, 172, 187]—also called Chimera methods—, sliding mesh techniques [11, 14], level-set methods [161], or standard arbitrary Lagrangian-Eulerian (ALE) formulations with frequent remeshing [143].

A different way to handle complex flow problems is to employ a comparably new and innovative class of methods, which enables the approximation of partial differential equations based on a set of nodes, without the need for an additional mesh. These meshfree methods (MMs) [21, 56] are able to solve problems where good meshes are difficult to maintain. The price to pay for this is the relatively high computational burden often associated with the use of these methods. Rather than calling these methods an alternative for meshbased methods, one may view them as specific tools, because they are frequently used in problems where meshes are difficult or impossible to maintain. A large number of meshfree methods have been developed during the last three decades, among them the popular element-free Galerkin (EFG) method [22] and smoothed particle hydrodynamics (SPH) [149]. For an overview of MMs see [21, 56].

The use of MMs for the simulation of challenging real-life problems is often inhibited by the increased computational cost, when compared with a meshbased simulation. This is particularly true for meshfree Galerkin methods, where a time-consuming integration

of the weak form under consideration is required. Therefore, MMs for the simulation of flow problems are usually applied in collocation settings, where the strong form of a problem is considered and no integration is needed. Advantages and disadvantages of meshfree collocation and Galerkin methods are well-known, and may be reduced to the statement that for the same number of unknowns Galerkin methods are in general more accurate and robust, but also computationally more demanding than collocation methods, see e.g. [15, 17, 37, 45].

Therefore, it seems promising to *couple* standard meshbased methods like the FEM with meshfree Galerkin methods. Then it is possible to use meshfree methods only in small regions of the domain where a mesh is difficult to maintain, and efficient meshbased standard methods in all other parts. Accuracy and robustness are expected to be favorable and the computational work is expected to scale with the meshbased part.

Rather than separating numerical methods into meshfree and meshbased, one may consider different formulations of the underlying differential equations. Most importantly Lagrangian, Eulerian, and ALE formulations [98] have to be considered which choose distinct coordinate systems for the description of the problem. The success of meshbased methods for the solution of flow problems is directly related to using Eulerian or ALE descriptions, because it seems impossible to maintain a suitable mesh for the Lagrangian formulation. Therefore, for the desired coupled meshfree/meshbased method it comes natural to employ the Eulerian or ALE viewpoint for the meshfree part of the domain as well. This is in contrast to the common practice to employ meshfree methods in Lagrangian collocation settings, i.e. as particle methods [112, 148, 149, 182].

An important difference between the formulations is in the presence of an advection term in the Eulerian and ALE formulations, which is absent in the Lagrangian description. Advection terms are non-selfadjoint operators that often lead to problems in their numerical treatment [31]. This is particularly the case for Bubnov-Galerkin methods, where the test functions are chosen equal to the shape functions. Spurious oscillations may pollute the overall solution and *stabilization* is required, see e.g. [31]. Furthermore, applying the same shape functions for all unknowns of the incompressible Navier-Stokes equations (equal-order-interpolation), which is from a computational viewpoint the most convenient way, leads to severe problems as a result from violating certain conditions. Again, stabilization is a possibility to overcome these problems [55]. The need for stabilization is well studied in the meshbased context, e.g. [31, 46, 97], and a number of stabilization methods have been developed. In this work, some of these methods are extended for the stabilization of

meshfree Galerkin methods, which is an important aspect for the success of the coupled meshfree/meshbased flow solver.

The coupling is realized by means of coupling meshfree and meshbased shape functions, rather than by combining meshfree and meshbased parts of the domain by Lagrange multipliers [79] or using physically motivated *ad hoc* approaches based on the conservation of mass and momentum [108]. For a coupling on shape-function-level, the domain is divided into a purely meshbased and meshfree part, separated by a layer of elements—called transition area—, where the coupling is realized. Standard approaches may be separated into those which consider the contribution of the finite element shape functions in the transition area in a modified computation of the meshfree shape functions [86], and those which employ monotone ramp functions in the transition area for a coupling [24]. It is found that in order to obtain shape functions that are suitable for a reliable stabilization, the original coupling approaches require modifications.

The resulting stabilized and coupled meshfree/meshbased flow solver is applied to complex fluid-structure interaction problems involving moving and rotating objects. Thereby test cases have been simulated which may not be solved by standard meshbased methods.

The plan of this work is as follows: Section 2 describes the model equations used throughout this work for the description of the fluid and the structure. It is started in 2.1 by pointing out some important principles for the description of different configurations in continuum mechanics, and Lagrangian, Eulerian, and ALE frameworks are described. Section 2.2 and 2.3 introduce St. Venant solids and Newtonian incompressible fluids as suitable models for the purposes of this work. The strong and weak form of the governing partial differential equations are shown on a continuous level, including suitable boundary conditions. In section 2.4, the coupling conditions between the fluid and the solid, as needed for the simulation of fluid-structure interaction phenomena, are specified.

In section 3, the discretization of the relevant model equations is discussed. The important principle of weighted residual methods is recalled in an abstract way in section 3.1. Sections 3.2 and 3.3 show the discretized weak forms and time integration schemes for the fluid and structure models. The resulting expressions are valid for finite element, meshfree, and coupled meshfree/meshbased shape functions. This section is ended in section 3.4 by a description of the partitioned, strongly coupled strategy employed for the simulation of fluid-structure problems.

The background of meshfree methods is worked out in section 4. Based on the report by

Fries and Matthies [56], a classification of the large number of different MMs is discussed. The moving least-squares (MLS) technique, being the fundamental principle in most MMs, is described in detail. Problems related to many MMs, like the imposition of essential boundary conditions, integration etc. are mentioned. Some of the most popular specific MMs are briefly described in section 4.4, and finally the MM chosen for this work is specified in detail.

An important step towards coupled meshfree/meshbased methods for flow problems is to establish stabilized meshfree Galerkin methods. The topic of stabilization is worked out in section 5, following conference contributions and reports by Fries and Matthies in [57, 60, 63, 65]. The description of the need for stabilization in meshbased as well as in meshfree methods, given in section 5.1, includes a review of various stabilization approaches. Standard stabilization methods, frequently used in the finite element context, are specified in section 5.2. It is found that the same stabilization approaches may directly be extended to MMs, however, the so-called stabilization parameter τ, which weighs the stabilization, requires certain attention. This is worked out in detail in section 5.3. The numerical results show the validity of the resulting stabilized meshfree Galerkin method, which may then be coupled with meshbased standard methods.

Following Fries and Matthies in [58, 59, 61, 62, 64, 66], coupling is discussed in section 6, starting with a description of several contexts in which coupling of meshfree and meshbased methods is desirable. Herein, coupling is performed in order to combine the efficiency of meshbased methods with the mesh-independency of MMs. Most importantly, two different coupling approaches exist for this purpose, which are described in section 6.3 and 6.4. It is found that modifications of these approaches are required in order to obtain shape functions that are more suitable for a reliable stabilization. The numerical results in section 6.5 validate the stabilized and coupled flow solver, including some test cases, which may not be solved by standard meshbased methods due to the presence of moving and rotating objects in the flow field.

Section 7 ends this work with an outlook and conclusion. In the appendix, conventions and symbols used throughout this work are described, and the meaning of all abbreviations is given.

2 Models in Continuum Mechanics

The aim of this work is to develop a coupled meshfree/meshbased method for the simulation of complex fluid-structure interaction problems. In this section, the well-known underlying partial differential equations for the description of the fluid and the structure are given for later reference. The standard models of continuum mechanics for an incompressible Newtonian fluid and a St. Venant solid are chosen, see e.g. [191]. The use of these simple standard models for the fluid and structure does not restrict the general applicability of the developed coupled method, and the application to other physical models—e.g. for neo-Hooke materials, plasticity, non-Newtonian fluids etc.—, is straightforward.

The natural way to state physical models in continuum mechanics is to formulate conservation laws. The resulting expressions are given in their weak form. Thereby, discontinuities are included naturally in the solution space. Assuming sufficiently smooth solutions allows to transform the models to the strong form, which is often understood more intuitively. Therefore, when describing the models for the fluid and structure, it is started with the strong form, introducing the weak form thereafter.

2.1 Eulerian, Lagrangian and ALE Formulations

Models in continuum mechanics may be formulated with respect to different reference systems. In this subsection, following [98, 162], Eulerian, Lagrangian and arbitrary Lagrangian-Eulerian (ALE) settings are briefly discussed.

A material domain $\Omega_{\boldsymbol{X}} \subset \mathbb{R}^d$ is considered, with d being the number of space dimensions. The motion of $\Omega_{\boldsymbol{X}}$ is described by a mapping $\boldsymbol{\phi} : \Omega_{\boldsymbol{X}} \times (0,T) \to \mathbb{R}^d$, and the image of $\Omega_{\boldsymbol{X}}$ at time t is denoted by the spatial domain $\Omega_{\boldsymbol{x}}$. A third domain $\Omega_{\boldsymbol{\chi}}$, the reference domain, is introduced and its motion is described by the mapping $\boldsymbol{\varphi} : \Omega_{\boldsymbol{\chi}} \times (0,T) \to \mathbb{R}^d$. Another mapping $\boldsymbol{\psi} : \Omega_{\boldsymbol{X}} \times (0,T) \to \mathbb{R}^d$, with $\boldsymbol{\psi} = \boldsymbol{\varphi}^{-1} \circ \boldsymbol{\phi}$ exists for sufficiently smooth, bijective functions $\boldsymbol{\phi}$ and $\boldsymbol{\varphi}$. See Fig. 1 for a schematic overview of the situation.

Let $\boldsymbol{x} \in \Omega_{\boldsymbol{x}}$, $\boldsymbol{X} \in \Omega_{\boldsymbol{X}}$, and $\boldsymbol{\chi} \in \Omega_{\boldsymbol{\chi}}$, then the following relations hold

$$\boldsymbol{x} = \boldsymbol{\phi}(\boldsymbol{X}, t), \quad \boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{\chi}, t), \quad \boldsymbol{\chi} = \boldsymbol{\psi}(\boldsymbol{X}, t), \tag{2.1}$$

$$\boldsymbol{d}_{\boldsymbol{\phi}} = \boldsymbol{x} - \boldsymbol{X}, \quad \boldsymbol{d}_{\boldsymbol{\varphi}} = \boldsymbol{x} - \boldsymbol{\chi}, \quad \boldsymbol{d}_{\boldsymbol{\psi}} = \boldsymbol{\chi} - \boldsymbol{X}, \tag{2.2}$$

where $\boldsymbol{d}$ are deformations; furthermore,

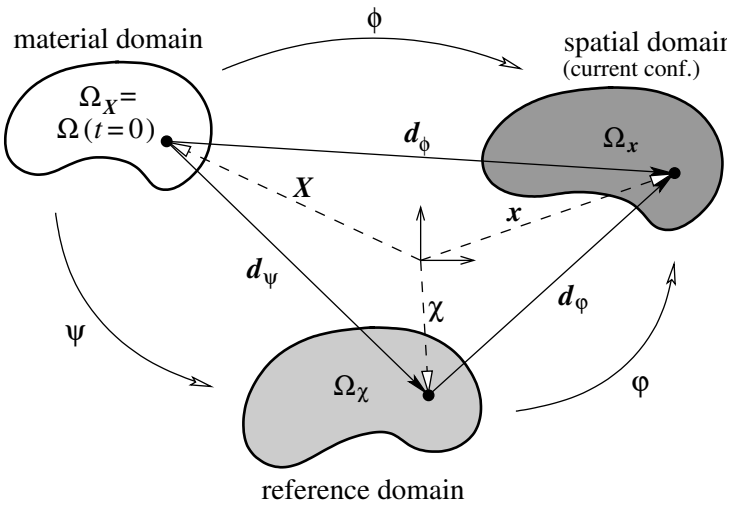

Figure 1: Different configurations of the continuum.

$$\begin{gathered}\dot{\boldsymbol{d}}_{\boldsymbol{\phi}} = \left.\frac{\partial \boldsymbol{d}_{\boldsymbol{\phi}}}{\partial t}\right|_{\boldsymbol{X}} = \left.\frac{\partial \boldsymbol{x}}{\partial t}\right|_{\boldsymbol{X}} = \dot{\boldsymbol{\phi}}, \quad \dot{\boldsymbol{d}}_{\boldsymbol{\varphi}} = \left.\frac{\partial \boldsymbol{d}_{\boldsymbol{\varphi}}}{\partial t}\right|_{\boldsymbol{\chi}} = \left.\frac{\partial \boldsymbol{x}}{\partial t}\right|_{\boldsymbol{\chi}} = \dot{\boldsymbol{\varphi}}, \\ \dot{\boldsymbol{d}}_{\boldsymbol{\psi}} = \left.\frac{\partial \boldsymbol{d}_{\boldsymbol{\psi}}}{\partial t}\right|_{\boldsymbol{X}} = \left.\frac{\partial \boldsymbol{\chi}}{\partial t}\right|_{\boldsymbol{X}} = \dot{\boldsymbol{\psi}}.\end{gathered} \tag{2.3}$$

Each function defined on any of the three domains implies the definitions of two other functions via composition

$$\begin{array}{ccccc} \underbrace{\boldsymbol{f}(\boldsymbol{\chi}, t)} & = & \underbrace{\boldsymbol{g}(\boldsymbol{x}, t)} & = & \underbrace{\boldsymbol{h}(\boldsymbol{X}, t)}, \\ \text{referential} & & \text{spatial} & & \text{material} \end{array} \tag{2.4}$$

so that by the chain rule for the derivative in time follows

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\chi}} \dot{\boldsymbol{d}}_{\boldsymbol{\psi}} + \frac{\partial \boldsymbol{f}}{\partial t} \quad = \quad \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}} \dot{\boldsymbol{d}}_{\boldsymbol{\phi}} + \frac{\partial \boldsymbol{g}}{\partial t} \quad = \quad \frac{\partial \boldsymbol{h}}{\partial t}, \tag{2.5}$$

where the relations (2.3) and $\frac{\partial \boldsymbol{X}}{\partial t} = 0$ are used. The time derivative of $\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{\psi}(\boldsymbol{X}, t), t)$ leads to

$$\dot{\boldsymbol{d}}_{\boldsymbol{\phi}} = \frac{\partial \boldsymbol{\varphi}}{\partial \boldsymbol{\psi}} \dot{\boldsymbol{d}}_{\boldsymbol{\psi}} + \dot{\boldsymbol{d}}_{\boldsymbol{\varphi}} \quad \Longrightarrow \quad \dot{\boldsymbol{d}}_{\boldsymbol{\psi}} = \frac{\partial \boldsymbol{\chi}}{\partial \boldsymbol{x}} \left(\dot{\boldsymbol{d}}_{\boldsymbol{\phi}} - \dot{\boldsymbol{d}}_{\boldsymbol{\varphi}} \right), \tag{2.6}$$

and inserting this into (2.5) gives

$$\begin{array}{ccccc} \underbrace{\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} \left(\dot{\boldsymbol{d}}_{\boldsymbol{\phi}} - \dot{\boldsymbol{d}}_{\boldsymbol{\varphi}} \right) + \dfrac{\partial \boldsymbol{f}}{\partial t}} & = & \underbrace{\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}} \dot{\boldsymbol{d}}_{\boldsymbol{\phi}} + \dfrac{\partial \boldsymbol{g}}{\partial t}} & = & \underbrace{\dfrac{\partial \boldsymbol{h}}{\partial t}}, \\ \text{referential} & & \text{spatial} & & \text{material} \end{array} \tag{2.7}$$

The important result is that a problem may be easily formulated in the ALE (reference) domain by still using the same spatial derivatives as in the purely Eulerian (spatial) setting and considering the difference in the convective velocity. In practice, this often results in the subtraction of the mesh (or node) velocity $\dot{\boldsymbol{d}}_{\boldsymbol{\varphi}} = \boldsymbol{u}_{\mathrm{M}}$ from the "real" advection velocity with respect to the material domain $\dot{\boldsymbol{d}}_{\boldsymbol{\phi}}$.

2.2 The Structure

A detailed description and analysis of different models in structural dynamics may be found for example in the books of Zienkiewicz and Taylor [191], and Bathe [12]. In this work, the models for Hooke solids and St. Venant solids are used. Both models are formulated in a total Lagrangian framework, i.e. all quantities of the solid are given with respect to the material configuration $\boldsymbol{X}$. They are suitable for sufficiently small strains and describe hyper-elastic materials. The difference between the two models is in the fact whether the model is able to incorporate large displacements and rotations (rigid body motions) or not, leading to a linear or non-linear formulation of the geometry.

The model for Hooke solids is only involved in the mesh-movement procedure, which is required in order to consider geometrical changes of the fluid domain, see section 3.4. In all other cases, the model for St. Venant solids is relevant.

St. Venant Solid. Before describing the strong and weak form of this model, some important well-known tensors in continuum mechanics are introduced. A fundamental measure of deformation is described by the material deformation gradient

$$\mathbf{F} = \nabla_{\boldsymbol{X}} \boldsymbol{x} = \nabla_{\boldsymbol{X}} \boldsymbol{d} + \mathbf{I}, \tag{2.8}$$

which maps a differential line segment in the material domain to the current configuration. $\mathbf{I}$ is the identity tensor. The (right) Cauchy-Green deformation tensor is introduced as

$$\mathbf{C} = \mathbf{F}^{\mathrm{T}} \mathbf{F}, \tag{2.9}$$

and the Green-Lagrange strain tensor $\mathbf{E}$ as

$$\mathbf{E} = \frac{1}{2}\left(\mathbf{C}-\mathbf{I}\right) = \frac{1}{2}\left(\mathbf{F}^{\mathrm{T}}\mathbf{F}-\mathbf{I}\right), \tag{2.10}$$

$$= \frac{1}{2}\left(\nabla_{\boldsymbol{X}}\boldsymbol{d} + \left(\nabla_{\boldsymbol{X}}\boldsymbol{d}\right)^{\mathrm{T}} + \left(\nabla_{\boldsymbol{X}}\boldsymbol{d}\right)^{\mathrm{T}}\left(\nabla_{\boldsymbol{X}}\boldsymbol{d}\right)\right). \tag{2.11}$$

An important property of this non-linear strain tensor is that rigid body motions do not generate strains. A suitable stress measure related to the Green-Lagrange strain tensor $\mathbf{E}$ is the second Piola-Kirchhoff stress tensor $\mathbf{S}$. It is computed by help of the stored energy function $W(\mathbf{E})$ in a hyper-elastic, homogenous, and isotropic material according to the St. Venant-Kirchhoff model

$$W_{\mathrm{SV}}(\mathbf{E}) = \frac{\lambda}{2}\left(\operatorname{tr}\mathbf{E}\right)^2 + \eta \operatorname{tr}\mathbf{E}^2 \tag{2.12}$$

as

$$\mathbf{S} = 2\frac{\partial W_{\mathrm{SV}}}{\partial \mathbf{C}} = \frac{\partial W_{\mathrm{SV}}}{\partial \mathbf{E}} = \lambda\left(\operatorname{tr}\mathbf{E}\right)\mathbf{I} + 2\eta\mathbf{E}. \tag{2.13}$$

It is noted that (2.12) and consequently also (2.13) only hold for small strains, but large displacements and rotations are considered correctly. λ and η are the Lamé constants. In engineering applications, the Lamé constants are often replaced by Young's modulus E and Poisson's ratio ν following the relation

$$\lambda = \frac{E\nu}{\left(1+\nu\right)\left(1-2\nu\right)}, \quad \eta = \frac{E}{2\left(1+\nu\right)}. \tag{2.14}$$

With these definitions, the strong form of the model for St. Venant solids in the material domain is [12, 191]

$$\varrho_{\mathrm{S}}\ddot{\boldsymbol{d}} - \nabla_{\boldsymbol{X}}\cdot\left(\mathbf{F}\mathbf{S}\right) = \boldsymbol{f}, \quad \text{on } \Omega_{\mathrm{S}}\times\left(0,T\right), \tag{2.15}$$

where ϱ_{S} is the density of the structure at the initial time step and $\boldsymbol{f}$ describes volume forces. The corresponding weak form may be stated as: Find $\boldsymbol{d}\in\left(\mathcal{H}^1_{\mathrm{E}}\right)^d$ such that

$$\int_{\Omega_{\mathrm{S}}}\varrho_{\mathrm{S}}\boldsymbol{w}\cdot\ddot{\boldsymbol{d}}\,d\Omega + \int_{\Omega_{\mathrm{S}}}\nabla_{\boldsymbol{X}}\boldsymbol{w} : \left(\mathbf{F}\mathbf{S}\right)\,d\Omega = \int_{\Omega_{\mathrm{S}}}\boldsymbol{w}\cdot\boldsymbol{f}\,d\Omega + \int_{\Gamma_{\mathrm{S},\boldsymbol{t}}}\boldsymbol{w}\cdot\hat{\boldsymbol{t}}\,d\Gamma \quad \forall\boldsymbol{w}\in\left(\mathring{\mathcal{H}}^1_{\mathrm{E}}\right)^d, \tag{2.16}$$

and is called principle of virtual work. The test functions $\boldsymbol{w}$ are considered as admissible virtual displacements $\delta\boldsymbol{d}$. The space $\mathcal{H}^1$ is the set of functions which are, together with

their first derivatives, square-integrable in Ω_{S}. The set $\mathcal{H}^1_{\mathrm{E}}$ consists of functions in $\mathcal{H}^1$ that take on the prescribed value for the displacement along the essential or Dirichlet boundary $\Gamma_{\mathrm{S},\boldsymbol{d}}$, and $\mathring{\mathcal{H}}^1_{\mathrm{E}}$ is the homogeneous counterpart such that the corresponding functions vanish along the Dirichlet boundary. It is mentioned that the second term in (2.16) is often written as $\int \delta\mathbf{E} : \mathbf{S}\, d\Omega$ instead of $\int \nabla_{\boldsymbol{X}} \boldsymbol{w} : (\mathbf{FS})\, d\Omega$, with $\delta\mathbf{E}$ being the variation of $\mathbf{E}$.

Hooke solid. For applications where not only the strains are sufficiently small but also the displacements, a simplified linear model may be chosen for the structure. Differences between the material and spatial configurations are neglected and a linear strain tensor $\mathbf{E}_{\mathrm{lin}}$ results as

$$\mathbf{E}_{\mathrm{lin}} = \frac{1}{2}\left(\nabla_{\boldsymbol{X}} \boldsymbol{d} + (\nabla_{\boldsymbol{X}} \boldsymbol{d})^{\mathrm{T}}\right). \tag{2.17}$$

The difference to the Green-Lagrange strain tensor $\mathbf{E}$ is in the absence of the non-linear third term in (2.11). The corresponding stress tensor is the Cauchy stress tensor $\mathbf{T}$. It follows from the strain energy function of a Hooke solid

$$W_{\mathrm{H}}\left(\mathbf{E}_{\mathrm{lin}}\right) = \frac{\lambda}{2}\left(\operatorname{tr}\mathbf{E}_{\mathrm{lin}}\right)^2 + \eta \operatorname{tr}\mathbf{E}^2_{\mathrm{lin}} \tag{2.18}$$

as

$$\mathbf{T} = \frac{\partial W_{\mathrm{H}}}{\partial \mathbf{E}_{\mathrm{lin}}} = \lambda\left(\operatorname{tr}\mathbf{E}_{\mathrm{lin}}\right)\mathbf{I} + 2\eta\mathbf{E}_{\mathrm{lin}}. \tag{2.19}$$

The resulting linear strong form for a structure undergoing small displacements and strains is [12, 191]

$$\varrho_{\mathrm{S}}\ddot{\boldsymbol{d}} - \nabla_{\boldsymbol{X}} \cdot \mathbf{T} = \boldsymbol{f}, \quad \text{on } \Omega_{\mathrm{S}} \times (0,T). \tag{2.20}$$

The similarity to (2.15) is obvious and the resulting weak form follows analogously to (2.16).

Boundary Conditions. The boundary is separated into three disjoint parts $\partial\Omega_{\mathrm{S}} = \Gamma_{\mathrm{S},\boldsymbol{d}} \bigcup \Gamma_{\mathrm{S},\boldsymbol{t}} \bigcup \Gamma_{\mathrm{c}}$. Dirichlet or essential boundary conditions for the solid are prescribed displacements $\hat{\boldsymbol{d}}$ on $\Gamma_{\mathrm{S},\boldsymbol{d}}$, i.e.

$$\boldsymbol{d} = \hat{\boldsymbol{d}}, \quad \text{on } \Gamma_{\mathrm{S},\boldsymbol{d}},\ \forall t \in (0,T). \tag{2.21}$$

Neumann or natural (traction) boundary conditions are applied on $\Gamma_{\mathrm{S},\boldsymbol{t}}$ as

$$\mathbf{S}\mathbf{F}^{\mathrm{T}}\boldsymbol{n} = \hat{\boldsymbol{t}}, \quad \text{on } \Gamma_{\mathrm{S},\boldsymbol{t}},\ \forall t \in (0,T)\,, \tag{2.22}$$

where $\boldsymbol{n}$ is the normal vector of the boundary. Γ_{c} is the coupling boundary with the fluid, and is discussed in section 2.4. The initial condition for the structure is

$$\boldsymbol{d}_0 = \boldsymbol{d}\left(\boldsymbol{X},0\right), \quad \dot{\boldsymbol{d}}_0 = \dot{\boldsymbol{d}}\left(\boldsymbol{X},0\right), \quad \text{on } \Omega_{\mathrm{S}},\ t=0. \tag{2.23}$$

2.3 The Incompressible Fluid

The formulation of the governing equations and an overview of standard methods for the approximation of the incompressible Navier-Stokes equations may be found e.g. in the books of Donea and Huerta [46], Ferziger and Perić [52], Gresho [70], and Löhner [139]. The instationary, incompressible Navier-Stokes equations are considered in velocity-pressure formulation. Let $\Omega_{\mathrm{F}} \subset \mathbb{R}^d$ be the spatial fluid domain in d dimensions, then these equations are in an arbitrary Lagrangian-Eulerian (ALE) framework [98]

$$\begin{aligned} \varrho_{\mathrm{F}}\left(\boldsymbol{u}_{,t} + \bar{\boldsymbol{u}}\cdot\nabla_{\boldsymbol{x}}\boldsymbol{u}\right) - \nabla_{\boldsymbol{x}}\cdot\boldsymbol{\sigma} - \boldsymbol{f} &= \boldsymbol{0}, \quad \text{on } \Omega_{\mathrm{F}}\times(0,T)\,, && (2.24)\\ \nabla_{\boldsymbol{x}}\cdot\boldsymbol{u} &= 0, \quad \text{on } \Omega_{\mathrm{F}}\times(0,T)\,, && (2.25) \end{aligned}$$

where (2.24) is the momentum equation, (2.25) the continuity equation, $\boldsymbol{u}$ are the velocities, and ϱ_{F} is the fluid density. It is mentioned that although the ALE framework is used, all operators and fluid quantities are formulated with respect to the spatial domain, by using the important result of (2.7). The advective velocity $\bar{\boldsymbol{u}} = \boldsymbol{u} - \boldsymbol{u}_{\mathrm{M}}$ takes movements of the reference ALE-coordinate system into account, $\boldsymbol{u}_{\mathrm{M}}$ is the velocity of the reference domain, i.e. $\dot{\boldsymbol{d}}_{\boldsymbol{\varphi}}$ in (2.7), which is in general the mesh velocity. The stress tensor $\boldsymbol{\sigma}$ of a Newtonian fluid is defined as

$$\boldsymbol{\sigma}\left(\boldsymbol{u},p\right) = -p\mathbf{I} + 2\mu\boldsymbol{\varepsilon}\left(\boldsymbol{u}\right), \text{ with } \boldsymbol{\varepsilon}\left(\boldsymbol{u}\right) = \frac{1}{2}\left(\nabla_{\boldsymbol{x}}\boldsymbol{u} + \left(\nabla_{\boldsymbol{x}}\boldsymbol{u}\right)^{\mathrm{T}}\right), \tag{2.26}$$

where p is the pressure, and μ the dynamic viscosity.

The weak form of the incompressible Navier-Stokes equations may be stated as follows [70]:

Find $\boldsymbol{u} \in \left(\mathcal{H}^1_{\mathrm{E}}\right)^d$ and $p \in L_2$ such that

$$\int_{\Omega_{\mathrm{F}}} \boldsymbol{w} \cdot \left[\varrho_{\mathrm{F}} \left(\boldsymbol{u}_{,t} + \bar{\boldsymbol{u}} \cdot \nabla_{\boldsymbol{x}} \boldsymbol{u}\right) - \boldsymbol{f}\right] d\Omega + \int_{\Omega_{\mathrm{F}}} \nabla_{\boldsymbol{x}} \boldsymbol{w} : \boldsymbol{\sigma}\left(\boldsymbol{u}, p\right) d\Omega + \int_{\Omega_{\mathrm{F}}} q \nabla_{\boldsymbol{x}} \cdot \boldsymbol{u} d\Omega = \int_{\Gamma_{\mathrm{F},\boldsymbol{h}}} \boldsymbol{w} \cdot \hat{\boldsymbol{h}} d\Gamma \qquad \forall \boldsymbol{w} \in \left(\mathring{\mathcal{H}}^1_{\mathrm{E}}\right)^d,\ \forall q \in L_2, \tag{2.27}$$

where $\left(\mathcal{H}^1_{\mathrm{E}}\right)^d$ and $\left(\mathring{\mathcal{H}}^1_{\mathrm{E}}\right)^d$ are sets of square-integrable functions with square-integrable first derivatives in Ω_{F}. The functions in $\left(\mathcal{H}^1_{\mathrm{E}}\right)^d$ take on the value of the prescribed velocities along the essential boundary $\Gamma_{\mathrm{F},\boldsymbol{u}}$, whereas those in $\left(\mathring{\mathcal{H}}^1_{\mathrm{E}}\right)^d$ vanish there. L_2 is the set of square-integrable functions in Ω_{F}. In (2.27), one frequently writes $\int \boldsymbol{\varepsilon}\left(\boldsymbol{w}\right) : \boldsymbol{\sigma}\left(\boldsymbol{u}, p\right) d\Omega$ instead of $\int \nabla_{\boldsymbol{x}} \boldsymbol{w} : \boldsymbol{\sigma}\left(\boldsymbol{u}, p\right) d\Omega$.

Boundary Conditions. The total boundary of the domain is divided into three complementary subsets $\partial\Omega = \Gamma_{\mathrm{F},\boldsymbol{u}} \bigcup \Gamma_{\mathrm{F},\boldsymbol{h}} \bigcup \Gamma_{\mathrm{c}}$, where on $\Gamma_{\mathrm{F},\boldsymbol{u}}$ Dirichlet boundary conditions for the velocity are prescribed, and on $\Gamma_{\mathrm{F},\boldsymbol{h}}$ Neumann (traction) boundary conditions are applied as

$$\boldsymbol{u} = \hat{\boldsymbol{u}} \quad \text{on } \Gamma_{\mathrm{F},\boldsymbol{u}},\ \forall t \in (0,T)\,, \tag{2.28}$$

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \hat{\boldsymbol{h}} \quad \text{on } \Gamma_{\mathrm{F},\boldsymbol{h}},\ \forall t \in (0,T)\,. \tag{2.29}$$

Γ_{c} is the coupling boundary with the structure and is further specified in the following subsection. The initial condition is given as a divergence-free velocity field

$$\boldsymbol{u}_0 = \boldsymbol{u}\left(\boldsymbol{x}, 0\right), \quad \nabla_{\boldsymbol{x}} \cdot \boldsymbol{u}_0 = 0, \quad \text{on } \Omega_{\mathrm{F}},\ t = 0. \tag{2.30}$$

2.4 Fluid-Structure Interface

The treatment of the interface between the fluid and the structure Γ_{c} is an important aspect in fluid-structure interaction problems. It should be noted that the fluid equations are stated in an ALE framework, whereas the structure equations are formulated in a pure Lagrangian framework. The consequence for the fluid domain is that Γ_{c} is a moving boundary, i.e. a function of time, whereas in the structure domain all variables and operators are formulated with respect to the initial situation, which means for the coupling interface with respect to $\Gamma^0_{\mathrm{c}} = \Gamma_{\mathrm{c}}\left(t = 0\right)$.

Along the coupling boundary, the displacements and velocities must coincide, i.e. $\forall \boldsymbol{\chi} \in \Gamma_{\mathrm{c}}(t)$, $\forall \boldsymbol{X} \in \Gamma_{\mathrm{c}}^{0}$

$$\boldsymbol{\chi} = \boldsymbol{X} + \boldsymbol{d}(\boldsymbol{X}, t), \tag{2.31}$$

$$\boldsymbol{u}(\boldsymbol{\chi}, t) = \dot{\boldsymbol{d}}(\boldsymbol{X}, t), \tag{2.32}$$

where the left hand side corresponds to the fluid domain and the right hand side to the structural domain. The way these conditions are enforced in the simulation of fluid-structure interaction problems is discussed in 3.4. The identity of the displacements is ensured by the adjustment of the fluid domain with respect to the structural domain, which, in practice, involves mesh-moving based on the ALE principle. The identity of the velocities is realized by adjusting the Dirichlet boundary conditions of the fluid along Γ_{c}. It is noted that from a physical viewpoint—and from (2.31) and (2.32)—also the accelerations along the coupling boundary should coincide. This, however, can often not be achieved from the numerical method used for the approximation, but the error introduced from this problem is negligible [150].

Another important coupling condition is that the traction from the fluid and structure balance each other along the fluid-structure interface,

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \frac{1}{\det \mathbf{F}} \mathbf{F}\mathbf{S}\mathbf{F}^{\mathrm{T}} \cdot \boldsymbol{n}. \tag{2.33}$$

Equation (2.33) shows the transformation of the stress tensor of the structure problem with respect to the initial situation onto the current deformed situation of the fluid domain.

3 Approximation Strategies

In the previous section some standard models in continuum mechanics for the fluid and structure were introduced. The exact solution of the partial differential equations (PDEs) involved in these models is limited to relatively simple problems. More complex systems may only be approximated in terms of various numerical methods. Most of these methods may be based on the principle of weighted residual methods which is briefly described in section 3.1. The method of weighted residuals introduces test and trial function for the approximation of the weak form of a PDE. Section 3.2 and 3.3 describe the *discretized* weak forms in case of the fluid and structure models discussed in the previous section. The test and trial functions are not further specified in this section. Thereby, the resulting expressions are generally applicable for approximations with either meshfree, meshbased or coupled meshfree/meshbased functions.

At the end of this section, the technique for coupling the discretized fluid and structure models in order to simulate fluid-structure interaction problems is described.

3.1 Weighted Residual Methods

Most numerical methods may be based on the principle of weighted residual methods, this method is briefly described here following [12]. In the weighted residual method, one is interested in finding an approximation of a function $u(\boldsymbol{x})$ of a PDE

$$\mathcal{L}u(\boldsymbol{x}) = f(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \Omega, \tag{3.1}$$

where $\mathcal{L}$ is a differential operator and f is the system's right hand side; suitable boundary conditions are assumed. An approximation u^h of the unknown field variable u is made through a linear combination of trial functions $\boldsymbol{N}$ (also called shape or ansatz functions) and unknown nodal parameters $\boldsymbol{u}$, hence

$$u(\boldsymbol{x}) \approx u^h(\boldsymbol{x}) = \boldsymbol{N}^{\mathrm{T}}(\boldsymbol{x})\,\boldsymbol{u} = \sum_{i=1}^{r} N_i(\boldsymbol{x})\,u_i. \tag{3.2}$$

It is always the aim to obtain the r unknown coefficients $\boldsymbol{u}$ such that a "good" approximation is obtained in a certain sense. Replacing u with u^h in the PDE gives $\mathcal{L}u^h - f^h = \varepsilon$. As it is in general not possible to fulfill the original PDE exactly with the approximation, a

residual error ε is introduced. The various weighted residual methods differ in the criteria that they employ to calculate $\boldsymbol{u}$ such that ε is small. In all techniques, $\boldsymbol{u}$ is determined in a way that a weighted average of ε vanishes. That is,

$$\int \boldsymbol{w}^h \varepsilon \, d\Omega = \int \boldsymbol{w}^h \left(\mathcal{L} u^h - f^h \right) \, d\Omega = \mathbf{0}, \tag{3.3}$$

where the functions $\boldsymbol{w}^h$ define the weighting of the residual and are called test (or weighting) functions. The weighted residual methods may be classified by the choice of the w_i, $i = 1, \ldots, r$:

- The **Bubnov-Galerkin method** results for $w_i = N_i$, all other choices are Petrov-Galerkin methods.
- **Collocation methods** result for $w_i = \delta \left(\boldsymbol{x} - \boldsymbol{x}_i \right)$, which is equivalent to setting ε to zero at r distinct points in the domain. The integral expressions in (3.3) vanish and the strong form of the PDE (3.1), evaluated at r distinct points, is obtained.
- **Subdomain collocation methods** often choose the test function $w_i = \chi_{\widetilde{\Omega}_i}$, where $\chi_{\widetilde{\Omega}_i}(\boldsymbol{x}) = 1$ for all $\boldsymbol{x} \in \widetilde{\Omega}_i \subset \Omega$ and vanishes otherwise is the characteristic function over certain distinct subdomains.
- The **least-squares method** is obtained if the test function is chosen such that the square of the residual is minimized with respect to $\boldsymbol{u}$, that is $\frac{\partial}{\partial \boldsymbol{u}} \int \varepsilon^2 \, d\Omega = 0$.
- If the test function is chosen to be the fundamental solution of the PDE, a **boundary element method** (or equivalent meshfree methods related with the boundary integral) result.

The integral expressions of the discrete weak form of the PDE (3.3) have to be evaluated—considering the given boundary conditions—with respect to $\boldsymbol{N}$ and $\boldsymbol{w}$. The resulting system of equations $\mathbf{A} \left(\boldsymbol{u} \right) = \boldsymbol{f}$ has to be solved for determining the unknowns $\boldsymbol{u}$.

It should be mentioned that one makes often use of the divergence theorem during this procedure to modify the integral expressions in order to shift continuity requirements between the trial and test functions.

The most popular representatives of weighted residual methods are the finite element and finite volume methods [12, 82]. They employ a mesh, which is part of the definition of

the shape and test functions. This is why they are considered members of the class of meshbased methods. All meshfree methods (MMs) may also be based on the weighted residual principle, see section 4.2, however, with the difference that they define shape and test functions without a mesh, only based on a set of nodes.

3.2 Discretization of the Structure

Different discretizations are made in space and time dimensions. In the time dimension, the Newmark and Hilber-Hughes-Taylor-α method are used here [81, 155]. Both methods are frequently used in practice for the simulation of dynamical structure problems. In the space dimension test and trial functions are used in a weighted residual setting for the approximation of the unknown displacements $\boldsymbol{d}$.

Discretization in Space. For the approximation of the displacements of a St. Venant or a Hooke solid the following test and trial function spaces $\mathcal{S}_{\boldsymbol{d}}^h$ and $\mathcal{V}_{\boldsymbol{d}}^h$ are introduced as

$$\mathcal{S}_{\boldsymbol{d}}^h = \left\{ \boldsymbol{d}^h \middle| \, \boldsymbol{d}^h \in \left(\mathcal{H}^{1h}\right)^d, \, \boldsymbol{d}^h = \hat{\boldsymbol{d}}^h \text{ on } \Gamma_{\mathrm{S},\boldsymbol{d}} \right\}, \tag{3.4}$$

$$\mathcal{V}_{\boldsymbol{d}}^h = \left\{ \boldsymbol{w}^h \middle| \, \boldsymbol{w}^h \in \left(\mathcal{H}^{1h}\right)^d, \, \boldsymbol{w}^h = \boldsymbol{0} \text{ on } \Gamma_{\mathrm{S},\boldsymbol{d}} \right\}, \tag{3.5}$$

where $\mathcal{H}^{1h}$ is a finite dimensional space built by the set of shape functions which are assumed to be elements of $\mathcal{H}^1$. Consequently, $\mathcal{H}^{1h}$ is a subspace of $\mathcal{H}^1$, i.e. $\mathcal{H}^{1h} \subseteq \mathcal{H}^1$. Then, the discretized weak form based on (2.16) may be formulated in the following Bubnov-Galerkin setting [191]: Find $\boldsymbol{d}^h \in \mathcal{S}_{\boldsymbol{d}}^h$ such that

$$\int_{\Omega_{\mathrm{S}}} \varrho_{\mathrm{S}} \boldsymbol{w}^h \cdot \ddot{\boldsymbol{d}}^h d\Omega + \int_{\Omega_{\mathrm{S}}} \nabla_{\boldsymbol{X}} \boldsymbol{w}^h : (\mathbf{FS}) \, d\Omega = \int_{\Omega_{\mathrm{S}}} \boldsymbol{w}^h \cdot \boldsymbol{f}^h d\Omega + \int_{\Gamma_{\mathrm{S},\boldsymbol{t}}} \boldsymbol{w}^h \cdot \hat{\boldsymbol{t}}^h d\Gamma \quad \forall \boldsymbol{w}^h \in \mathcal{V}_{\boldsymbol{d}}^h, \tag{3.6}$$

where $\hat{\boldsymbol{t}}^h$ characterizes the traction along the Neumann boundary. In the linear case of a Hooke solid, (**FS**) in (3.6) is replaced by **T**.

Discretization in Time. A number of different time integration methods may be chosen in order to discretize the semi-discrete system of equations arising from (3.6), which is of the form

$$\mathbf{M}\ddot{\boldsymbol{d}}_{n+1}^h + \mathbf{C}\dot{\boldsymbol{d}}_{n+1}^h + \mathbf{K}\left(\boldsymbol{d}_{n+1}^h\right) = \boldsymbol{f}\left(t_{n+1}\right). \tag{3.7}$$

In case of a Hooke solid, the stiffness matrix $\mathbf{K}$ is linear and one may write $\mathbf{K}\boldsymbol{d}^h_{n+1}$ instead of $\mathbf{K}\left(\boldsymbol{d}^h_{n+1}\right)$. The damping matrix $\mathbf{C}$ is zero in case of (3.6) but is considered here for completeness, so that damping effects could easily be incorporated. See e.g. [12, 18, 188, 191] for an overview of various time discretization methods. In this work, two methods are employed. One is the Newmark method [155], which is defined as

$$\boldsymbol{d}^h_{n+1} = \boldsymbol{d}^h_n + \Delta t \dot{\boldsymbol{d}}^h_n + \frac{\Delta t^2}{2}\left[(1-2\beta)\,\ddot{\boldsymbol{d}}^h_n + 2\beta \ddot{\boldsymbol{d}}^h_{n+1}\right], \tag{3.8}$$

$$\dot{\boldsymbol{d}}^h_{n+1} = \dot{\boldsymbol{d}}^h_n + \Delta t\left[(1-\gamma)\,\ddot{\boldsymbol{d}}^h_n + \gamma \ddot{\boldsymbol{d}}^h_{n+1}\right]. \tag{3.9}$$

(3.7), (3.8) and (3.9) may be arranged in a way that a system of equation for $\boldsymbol{d}^h_{n+1}$ is obtained, from which, in a post-processing step, $\dot{\boldsymbol{d}}^h_{n+1}$ and $\ddot{\boldsymbol{d}}^h_{n+1}$ are computed. The parameters β and γ characterize the accuracy and stability properties of the resulting Newmark method. Accuracy of second order in time is obtained for $\gamma = 1/2$. Together with a choice of $\beta = 1/4$, the method is implicit and for *linear* problems unconditionally stable, i.e. it is stable for any time-step. Furthermore, the total energy of the system is maintained exactly in this case, that is, no artificial damping effects occur. In the geometrically non-linear case of the model for St. Venant solids, the energy conservation and stability properties of the Newmark method are different compared to the linear case of a Hooke solid [155]. Then, in the absence of physical damping effects, the Newmark method may show poor stability properties. The stability properties often improve for $\gamma > 1/2$ due to an increased numerical damping, however, second order accuracy is no longer achievable then.

The second realized time discretization method is the Hilber-Hughes-Taylor-α (HHT-α) method [81], which is a generalization of the Newmark method. It allows numerical damping, adjustable by the parameter α, without giving up second order accuracy. For $\alpha = 0$, the Newmark method is recovered. The system of equations arising from (3.6) is of the form

$$\mathbf{M}\ddot{\boldsymbol{d}}^h_{n+1} + \mathbf{C}\dot{\boldsymbol{d}}^h_{n+1+\alpha} + \mathbf{K}\left(\boldsymbol{d}^h_{n+1+\alpha}\right) = \boldsymbol{f}\left(t_{n+1+\alpha}\right), \tag{3.10}$$

with

$$\boldsymbol{d}^h_{n+1+\alpha} = (1+\alpha)\,\boldsymbol{d}^h_{n+1} - \alpha \boldsymbol{d}^h_n, \tag{3.11}$$

$$\dot{\boldsymbol{d}}^h_{n+1+\alpha} = (1+\alpha)\,\dot{\boldsymbol{d}}^h_{n+1} - \alpha \dot{\boldsymbol{d}}^h_n, \tag{3.12}$$

$$t_{n+1+\alpha} = (1+\alpha)\,t_{n+1} - \alpha t_n. \tag{3.13}$$

The other two equations of the Newmark method, (3.8) and (3.9), remain. A second order accurate scheme with favorable stability properties is obtained for $-1/3 \leq \alpha \leq 0$, $\beta = (1-\alpha)^2/4$, and $\gamma = (1-2\alpha)/2$.

3.3 Discretization of the Incompressible Fluid

The discretization of the fluid is—similarily to the structure—separated into the space and time dimensions. In the time direction, the Crank-Nicholson method (trapazoidal rule) is employed, whereas in the space dimension test and trial functions are chosen in a weighted residual setting for the approximation of the unknown velocities $\boldsymbol{u}$ and pressure p.

Discretization in Space. For a Bubnov-Galerkin discretization with equal-order interpolation for all unknowns, the test and trial functions for the velocities and pressure are from the sets

$$\mathcal{S}_{\boldsymbol{u}}^h = \left\{ \boldsymbol{u}^h \middle| \, \boldsymbol{u}^h \in \left(\mathcal{H}^{1h}\right)^d, \, \boldsymbol{u}^h = \hat{\boldsymbol{u}}^h \text{ on } \Gamma_{\mathrm{F},\boldsymbol{u}} \right\}, \tag{3.14}$$

$$\mathcal{V}_{\boldsymbol{u}}^h = \left\{ \boldsymbol{w}^h \middle| \, \boldsymbol{w}^h \in \left(\mathcal{H}^{1h}\right)^d, \, \boldsymbol{w}^h = \boldsymbol{0} \text{ on } \Gamma_{\mathrm{F},\boldsymbol{u}} \right\}, \tag{3.15}$$

$$\mathcal{S}_p^h = \mathcal{V}_p^h = \left\{ q^h \middle| \, q^h \in \mathcal{H}^{1h} \right\}, \tag{3.16}$$

where $\mathcal{H}^{1h}$ is described in the previous subsection. It is pointed out that from a mathematical viewpoint it would be sufficient to choose q^h to be a member of the L_2-space, however, equal-order interpolation of the velocities and pressure is employed in this work, as described in detail in section 5. The discretized weak form of the *unstabilized* incompressible Navier-Stokes equations in ALE formulation may be stated based on (2.27) as follows [178]: find $\boldsymbol{u}^h \in \mathcal{S}_{\boldsymbol{u}}^h$ and $p^h \in \mathcal{S}_p^h$ such that

$$\begin{aligned} &\int_{\Omega_{\mathrm{F}}} \boldsymbol{w}^h \cdot \left[\varrho_{\mathrm{F}} \left(\boldsymbol{u}_{,t}^h + \bar{\boldsymbol{u}}^h \cdot \nabla_{\boldsymbol{x}} \boldsymbol{u}^h \right) - \boldsymbol{f}^h \right] d\Omega + \int_{\Omega_{\mathrm{F}}} \nabla_{\boldsymbol{x}} \boldsymbol{w}^h : \boldsymbol{\sigma}^h \left(\boldsymbol{u}^h, p^h \right) d\Omega + \\ &\int_{\Omega_{\mathrm{F}}} q^h \nabla_{\boldsymbol{x}} \cdot \boldsymbol{u}^h d\Omega = \int_{\Gamma_{\mathrm{F},\boldsymbol{h}}} \boldsymbol{w}^h \cdot \hat{\boldsymbol{h}}^h d\Gamma \qquad \forall \boldsymbol{w}^h \in \mathcal{V}_{\boldsymbol{u}}^h, \, \forall q^h \in \mathcal{V}_p^h. \end{aligned} \tag{3.17}$$

It is important to note that this unstabilized standard Bubnov-Galerkin approximation with equal-order interpolation may *not* be used in this way in order to obtain suitable results. In fact, stabilization is required, which is an important aspect of this work, and is discussed in detail in section 5.

Discretization in Time. Time integration is performed with a Crank-Nicholson scheme, see e.g. [46],

$$\boldsymbol{u}_{,t}^h = \frac{\boldsymbol{u}_{n+1}^h - \boldsymbol{u}_n^h}{\Delta t}, \quad \boldsymbol{u}^h = \theta \boldsymbol{u}_{n+1}^h + (1-\theta)\, \boldsymbol{u}_n^h, \quad \theta = \frac{1}{2}, \tag{3.18}$$

which may be seen as a certain version of the theta-method [46]. The velocities $\boldsymbol{u}_n^h$ are related to the previous time step. As there is no time-derivative in the continuity equation (2.25), the term $\int_{\Omega_\mathrm{F}} q^h \nabla_{\boldsymbol{x}} \cdot \boldsymbol{u}^h d\Omega$ becomes directly $\int_{\Omega_\mathrm{F}} q^h \nabla_{\boldsymbol{x}} \cdot \boldsymbol{u}_{n+1}^h d\Omega$. The pressure is computed fully implicitly [139], i.e. p^h in (3.17) may directly be replaced by p_{n+1}^h. The resulting formulation is solved for the unknowns of the current time step $\left(\boldsymbol{u}_{n+1}^h, p_{n+1}^h\right)$. This time-integration is second order accurate, $O\left(\Delta t^2\right)$, and possesses favorable stability properties [46, 70]. Other frequently used time-integration schemes are for example first-order explicit schemes, Runge-Kutta schemes, discontinuous Galerkin methods etc., see e.g. [46, 70, 139, 168]. All these methods differ in their numerical properties, most importantly accuracy and stability, and the computational effort associated with their use.

3.4 A Partitioned, Strongly Coupled Method for Fluid-Structure Interaction

Simulation of fluid-structure interaction problems involves the solution of the fluid and structure models with respect to the boundary and coupling conditions. Furthermore, the fluid domain moves according to the deformation of the structure, and is thereby part of the solution itself. The adaption of the fluid domain is realized by using the ALE framework for the description of the fluid model, and a mesh-movement procedure is involved. In contrast, the structure problem is formulated in a purely Lagrangian framework and is solved with respect to the initial mesh in every time step. It is pointed out that by using the standard term "mesh-movement" no restriction to meshbased methods is intended. The change of the problem domains is in fact realized by a *node* movement which is applicable to meshfree methods as well.

The coupling in time is realized by a partitioned method with strong coupling [144, 150]. The fluid and structure problems are solved individually and coupling is done by a Neumann-to-Dirichlet principle. Other strategies for the coupling in time, see [144, 150] for an overview, differ in the treatment of the coupling conditions or the way they perform the iteration of the fluid-structure loop in every time step.

A description of the partitioned strong coupling algorithm is as follows, see also Fig. 2: In each time step, an iteration loop over the fluid and the structure domain is performed until convergence is reached. The first step of the iteration loop is the approximation of the flow field. This gives the traction $\boldsymbol{\sigma} \cdot \boldsymbol{n}$ along the fluid-structure interface Γ_{c}, which is used as the loading (Neumann boundary) for the structure problem according to (2.33). The solution of the structure problem is the next step and leads to displacements $\boldsymbol{d}$ of the structure along Γ_{c}. The fluid mesh has to be modified in a way such that the deformations of the structure are mapped correctly. That is, the fluid interface at the current time step must match the structure mesh of the initial time step plus the current deformation. This is realized by a mesh-moving algorithm, which is the third step of the fluid-structure iteration loop.

There are several ways to achieve a suitable, conforming fluid mesh. Explicit algebraic expressions may be used in each direction of the domain [74], or the spring analogy may be employed [27], where element edges are replaced by linear springs. Furthermore, sub-problems in the fluid domain may be constructed, which is the technique chosen in this work. For example, a pseudo-structure problem of the fluid domain may be solved on $\Omega_{\mathrm{PS}} \subseteq \Omega_{\mathrm{F}}$ [13, 166]. For the sake of simplicity, the pseudo-structure may be described by the stationary version of the model for Hooke solids. The Dirichlet conditions are the displacements of the real structure along Γ_{c} and 0 at all other parts of the boundary $\partial\Omega_{\mathrm{PS}}$. Alternatively, in this context also the Laplace equation may be solved for each component of $\boldsymbol{d}$, see e.g. [143, 162],

$$\Delta d_i = \nabla \cdot \nabla d_i = 0, \quad \text{on } \Omega_{\mathrm{M}},\ i = 1, \ldots, d. \tag{3.19}$$

After the update of the node positions of the fluid mesh, the mesh (or node) velocity $\boldsymbol{u}_{\mathrm{M}}$ is computed as

$$\boldsymbol{u}_{\mathrm{M}} = \frac{\boldsymbol{x}_{i+1} - \boldsymbol{x}_n}{\Delta t}, \tag{3.20}$$

where $\boldsymbol{x}_n$ are the node positions in Ω_{F} of the previous time step, and $\boldsymbol{x}_{i+1}$ are the node positions of the current step in the fluid-structure iteration loop ($\boldsymbol{x}_{n+1} = \boldsymbol{x}_{i+1}$ if the stopping criteria of this loop is met). The Dirichlet boundary conditions for the flow velocities along the coupling interface Γ_{c} have to be adjusted such that they take the mesh velocity into account. The new node positions, node velocities and adjusted Dirichlet boundary conditions are the input data for the next fluid step. This iteration is performed in every time step until convergence is reached. Throughout this work, the convergence

of the fluid-structure iteration loop was in none of the test cases critical, see e.g. [150] for further remarks on the robustness and other numerical properties of this strongly coupled strategy for the simulation of fluid-structure interaction problems.

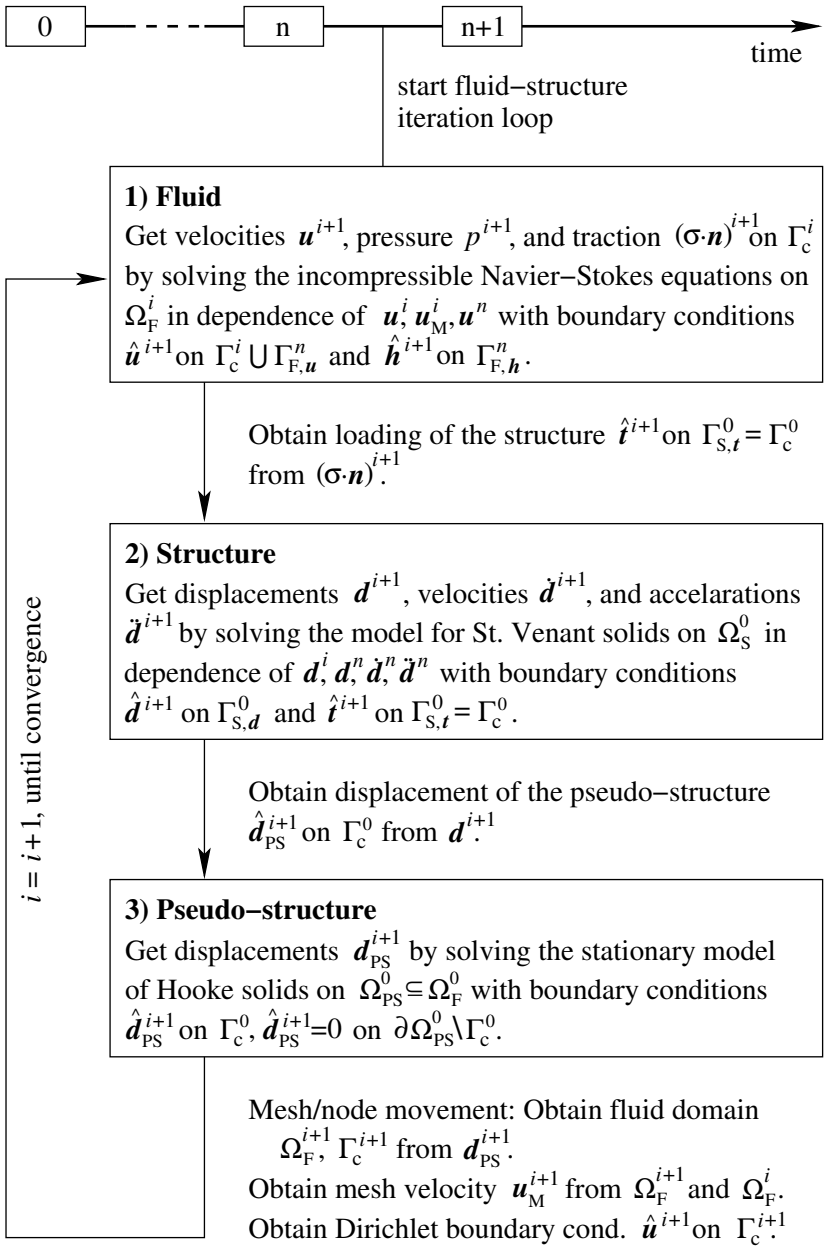

Figure 2: Fluid-structure iteration loop of the partitioned, strongly coupled method in every time step.

4 Meshfree Methods

Conventional numerical methods need an *a priori* definition of the connectivity of the nodes, i.e. they rely on a mesh. The finite element method (FEM) and finite volume method (FVM) may be the most well-known members of these thoroughly developed meshbased methods, see e.g. [12, 82, 92, 191]. In contrast, *meshfree methods* (MMs) approximate partial differential equations only based on a set of nodes, without the need for an additional mesh. A multitude of different MMs has been published during the last three decades. Despite the variety of names, origins, and viewpoints of individual methods it is interesting to note that in fact there are significant similarities between many of these methods.

Surveys on MMs may for example be found in [21] by Belytschko *et al.*, in [56] by Fries and Matthies, and in [125] by Li and Liu. Special issues of journals on various aspects of MMs may be found in [34, 35, 132]; some books on MMs are also available, see e.g. [4, 72, 73, 126, 130].

In section 4.1, general features of most MMs are mentioned. Then, following Fries and Matthies [56], MMs are put into a unified context and a classification and overview of MMs is given. The MLS procedure, being a fundamental principle for many MMs, is worked out in detail in section 4.3, and related problems to many MMs are discussed. Some specific meshfree methods from the class of collocation and Bubnov-Galerkin methods are described in section 4.4. Finally, the choice of the particular meshfree method used throughout this work is described and justified.

4.1 General Features of Meshfree Methods

Some of the most important general features of most MMs are listed in the following, often comparing them with the analogous properties of meshbased methods:

- Absence of a mesh
  - In MMs the connectivity of the nodes is determined at run-time.
  - No mesh alignment sensitivity. This is a serious problem in meshbased calculations e.g. of cracks and shear bands [125].
  - h-adaptivity is comparably simple with MMs as only nodes have to be added, and the connectivity is then computed at run-time automatically. p-adaptivity is also conceptionally simpler than in meshbased methods [21].

  - No mesh generation at the beginning of the calculation is necessary. This is in general still not a fully automatic process, especially not in complex three-dimensional domains, and may require major human interactions [104].
  - No remeshing during the calculation. Especially in problems with large deformations of the domain or moving discontinuities, a frequent remeshing is often needed in meshbased methods, however, a conforming mesh with sufficient quality may be impossible to maintain. Even if it is possible, the remeshing process degrades the accuracy considerably due to the perpetual projection between the meshes [21], and the post-processing in terms of visualization and time-histories of selected points requires a large effort [23].

- Continuity of shape functions: The shape functions of MMs may easily be constructed to have any desired order of continuity.

  - MMs readily fulfill the requirement on the continuity arising from the order of the problem under consideration. In contrast, in meshbased methods the construction of even C^1-continuous shape functions—needed e.g. for the solution of forth order boundary value problems—may pose a serious problem [127].
  - No post-processing is required in order to determine smooth derivatives of the unknown functions, e.g. smooth strains.
  - Special cases where the continuity of the meshfree shape functions and derivatives is not desirable, e.g. in cases where physically justified discontinuities like cracks, different material properties etc. exist, can be handled with certain techniques.

- Convergence: For the same order of consistency numerical experiments suggest that the convergence results of the MMs are often considerably better than the results obtained by meshbased shape functions [122]. However, theory fails to predict this higher order of convergence [122].

- Computational effort: In practice, for a given reasonable accuracy, MMs are often considerably more time-consuming than their meshbased counterparts.

  - Meshfree shape functions are of a more complex nature than the polynomial-like shape functions of meshbased methods. Consequently, the number of integration

points for a sufficiently accurate evaluation of the integrals of the weak form is considerably larger in MMs than in meshbased methods. In collocation MMs no integration is required, however, this advantage is often compensated by evoking accuracy and stability problems.

  - At each integration point the following steps are often necessary to evaluate meshfree shape functions: Neighbour search, solution of small systems of equations and small matrix-matrix and matrix-vector operations in order to determine the derivatives.
  - The resulting global system of equations has in general a larger bandwidth for MMs than for comparable meshbased methods [21].

- Essential (Dirichlet) boundary conditions: Most MMs lack the Kronecker-δ property, i.e. the meshfree shape functions N_i do not fulfill $N_i(\boldsymbol{x}_j) = \delta_{ij}$. This is in contrast to meshbased methods which often have this property. Consequently, the imposition of essential boundary conditions requires certain attention in MMs and may degrade the convergence of the method [77].

- Locking: Differently from what has been stated in early papers [22] it should be mentioned that MMs may as well suffer from the locking phenomenon, similarly to the FEM, see [38, 90]. It is sometimes possible to alleviate this phenomenon by tuning some parameters of the MM.

4.2 Classification of Meshfree Methods

Before focusing on *one* specific MM, which is well-suited for the purpose of simulating complex flow phenomena, a classification of the large number of different MMs is discussed. With this background one may then justify the specific choice of a MM used herein. Three identifying criteria have been found in [56] by Fries and Matthies for the classification of the various MMs:

- Construction of a set of functions that build a partition of unity (PU) of n-th order.

- Choice of an approximation in a weighted residual setting, which can use the PU functions directly as shape functions, or enhance the approximation properties by an additional extrinsic basis.

- Choice of the test functions in a weighted residual setting.

4.2.1 Partition of Unity and Consistency

Before discussing these three steps in the classification of MMs, the term partition of unity is introduced. A set of r functions $\{\boldsymbol{N}\}$ builds a partition of unity of order n over a d-dimensional domain $\Omega \subseteq \mathbb{R}^d$ if

$$\sum_{i=1}^{r} N_i(\boldsymbol{x})\, \boldsymbol{p}(\boldsymbol{x}_i) = \boldsymbol{p}(\boldsymbol{x}) \quad \forall \boldsymbol{x} \in \Omega, \tag{4.1}$$

where $\boldsymbol{x}_i$ are the corresponding nodes of the functions N_i, and $\boldsymbol{p}$ forms a basis of the approximating subspace, which in general consists of monomials. A convenient way to formulate a useful basis is possible with help of the multi-index notation. The multi-index $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d)$ is used in the following with $\alpha_i \in \mathbb{N}_0^+$. If $\boldsymbol{\alpha}$ is applied to a vector $\boldsymbol{x}$ of the same length then

$$\boldsymbol{x}^{\boldsymbol{\alpha}} = x_1^{\alpha_1} \cdot x_2^{\alpha_2} \cdots x_d^{\alpha_d}, \tag{4.2}$$

and $D^{\boldsymbol{\alpha}} u(\boldsymbol{x})$ is the $\boldsymbol{\alpha}$-th Fréchet derivative [190] of the function u, that is

$$D^{\boldsymbol{\alpha}} u(\boldsymbol{x}) = \frac{\partial^{|\boldsymbol{\alpha}|} u(\boldsymbol{x})}{\partial^{\alpha_1} x_1 \partial^{\alpha_2} x_2 \cdots \partial^{\alpha_d} x_d}. \tag{4.3}$$

The length of $\boldsymbol{\alpha}$ is $|\boldsymbol{\alpha}| = \sum_{i=1}^{d} \alpha_i$. With this notation one can easily define a polynomial basis of order n as

$$\boldsymbol{p}(\boldsymbol{x}) = \left\{ \boldsymbol{x}^{\boldsymbol{\alpha}} \middle| \, |\boldsymbol{\alpha}| \leq n \right\}. \tag{4.4}$$

For example, a quadratic basis ($n = 2$) in one dimension ($d = 1$) yields $\boldsymbol{p}^{\mathrm{T}} = [1,\, x,\, x^2]$ and a linear basis in two dimensions $\boldsymbol{p}^{\mathrm{T}} = [1,\, x,\, y]$. The relationship between the dimension d and the order n on the one hand and the number of components k in the basis vector $\boldsymbol{p}$ on the other hand is

$$k = \frac{1}{d!} \prod_{i=1}^{d} (n + i). \tag{4.5}$$

The set $\left\{ \boldsymbol{\alpha} \middle| \, |\boldsymbol{\alpha}| \leq n \right\}$ may be interpreted as a $k \times d-$matrix, where the i-th line of this matrix is denoted by the multi-index $\boldsymbol{\alpha}^i$—whereas, in contrast, α_i is a component of a certain multi-index. It is assumed that the multi-indices $\boldsymbol{\alpha}^i$ in $\left\{ \boldsymbol{\alpha} \middle| \, |\boldsymbol{\alpha}| \leq n \right\}$ are sorted such that $|\boldsymbol{\alpha}^i| \leq |\boldsymbol{\alpha}^j|$ for all $i \neq j$ and $1 \leq i, j \leq k$.

Then, a PU of order $n \geq 0$ fulfills at least $\sum_i N_i(\boldsymbol{x}) = 1$ as, according to (4.4), $p_1(\boldsymbol{x}) = \boldsymbol{x}^{\boldsymbol{\alpha}^1} = \boldsymbol{x}^{(0,\ldots,0)} = 1$, which reflects the basic meaning of the term “partition of unity”.

In a weighted residual method, see section 3.1, having an approximation of the form $u \approx u^h = \boldsymbol{N}^{\mathrm{T}}\boldsymbol{u} = \sum_i N_i u_i$, where $\{\boldsymbol{N}\}$ builds a PU of order n, it is possible to reproduce a polynomial solution of a PDE under consideration up to this order *exactly* [7, 8]. There is a strong relation to the order of *consistency* of the resulting approximation. In a mathematical sense, a scheme $\mathcal{L}^h u^h = f^h$ is consistent of order n with the differential equation $\mathcal{L}u = f$, if the approximation error

$$\left\| \mathcal{L}u - \mathcal{L}^h u^h \right\| = O\left(h^n\right), \tag{4.6}$$

where h is some measure of the node distribution. Using a PU of order n in a weighted residual approximation for u leads to an $n-$th order consistent method [21]. Therefore, (4.1) is also called consistency condition or reproducing condition of order n.

The reproducing conditions for the derivatives of the functions N_i follow immediately from (4.1) as

$$\sum_{i=1}^{r} D^{\boldsymbol{\alpha}} N_i\left(\boldsymbol{x}\right) \, \boldsymbol{p}\left(\boldsymbol{x}_i\right) = D^{\boldsymbol{\alpha}} \boldsymbol{p}\left(\boldsymbol{x}\right). \tag{4.7}$$

Depending on the order of the PDE under consideration, there are certain consistency requirements. For example, approximating a PDE of order $2n$ with a Galerkin method (where the weak form is considered) requires test and shape functions with n-th order consistency.

4.2.2 Step 1: Construction of a Partition of Unity

In the previous subsection the term PU has been defined. Given a domain Ω, there are infinitely many PUs that fulfill (4.1) with r functions. A major difference is whether a PU is constructed with the help of a mesh or not.

Standard finite element shape functions build a PU of order n, and the supports of these functions are defined by means of a mesh. As an example, the quadratic finite element shape functions N_i are considered in a one-dimensional element Ω_{e}, see Fig. 3, with nodes x_1, x_2, x_3:

$$N_i\left(x\right) = \frac{x^2 - \left(x_j + x_k\right) x + x_j \cdot x_k}{\left(x_k - x_i\right)\left(x_j - x_i\right)}\cdot, \quad \forall i,j,k : \epsilon_{ijk} = 1, \tag{4.8}$$

where ϵ_{ijk} is the permutation symbol. Algebraic manipulations directly show that (4.1)

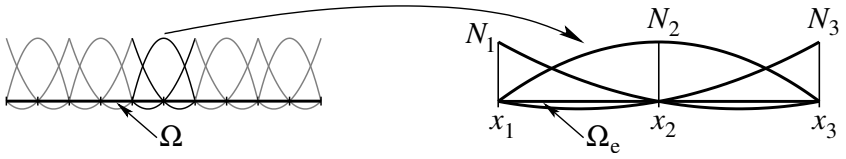

Figure 3: Quadratic FE shape functions build a meshbased PU of order 2.

applied to this situation

$$\sum_{i=1}^{3} N_i(x) \begin{pmatrix} 1 \\ x_i \\ x_i^2 \end{pmatrix} = \begin{pmatrix} 1 \\ x \\ x^2 \end{pmatrix}, \quad \forall \boldsymbol{x} \in \Omega_e \tag{4.9}$$

is fulfilled. That is, the quadratic shape functions build a PU of order 2 over the one-dimensional element, which may be directly extended for any one-dimensional domain discretized by several quadratic elements. It is emphasized that these shape functions are defined based on the one-dimensional element, i.e. with the help of a mesh.

There are different *meshfree* ways to construct PUs of a certain order. Most MMs rely on the fundamental moving least-squares (MLS) technique, introduced by Lancaster and Salkauskas [121]. This procedure is discussed in detail in section 4.3, where also the equivalence to the reproducing kernel particle methods (RKPM) is pointed out. In the MLS procedure, based on a set of nodes and a related weighting function with compact support, PUs of any order may be constructed in conceptionally the same way.

There are other MMs using different ways for the construction of a PU. Sibson [170] and non-Sibson [16] interpolation functions build PUs, which are used in the natural element method (NEM) [29, 175, 176] and in the meshless finite element method (MFEM) [103, 104]. The reproducing kernel element method (RKEM), recently introduced in [135, 127, 140, 171] constructs arbitrarily smooth element shape functions over elements which also fulfill the PU-conditions. Also coupled meshfree/meshbased functions may be constructed such that they build a PU of a certain order over Ω. This will further be discussed in section 6.

4.2.3 Step 2: Choice of an Approximation

In a weighted residual method, the unknown field variable u is approximated by a linear combination of shape functions $\boldsymbol{N}$ with unknown coefficients $\boldsymbol{u}$, i.e.

$$u^h(\boldsymbol{x}) = \boldsymbol{N}^{\mathrm{T}}(\boldsymbol{x})\,\boldsymbol{u} = \sum_{i=1}^{r} N_i(\boldsymbol{x})\,u_i. \tag{4.10}$$

If the shape functions $\boldsymbol{N}$ fulfill the PU-conditions of order n, (4.1), then the resulting method will be consistent of order n, assuming that other requirements are fulfilled (such as the minimum requirement for the continuity and the order of consistency related to the PDE under consideration). On this basis, a broad theoretical background has been developed in [7, 8], and the classical FEM builds a subcase of this approach.

It is also possible to extend the approximation space with an additional extrinsic basis $\boldsymbol{q}$. Thereby, the approximation space may be enriched by specific functions or the order of consistency may be improved indirectly. Typical methods choose an approximation of

$$u^h(\boldsymbol{x}) = \sum_{i=1}^{r} N_i(\boldsymbol{x})\,\boldsymbol{q}^{\mathrm{T}}(\boldsymbol{x})\,\boldsymbol{v}_i, \tag{4.11}$$

or

$$u^h(\boldsymbol{x}) = \sum_{i=1}^{r} N_i(\boldsymbol{x})\left(u_i + \boldsymbol{q}^{\mathrm{T}}(\boldsymbol{x})\,\boldsymbol{v}_i\right). \tag{4.12}$$

Each $\boldsymbol{v}_i$ is a vector of unknowns. Obviously, if l is the number of components in the basis $\boldsymbol{q}$, then there are l times more unknowns from the approximations (4.11) and (4.12) when compared with (4.10). One example for an application of this idea is to extend a zero-order PU to a first order consistent approximation by choosing the extrinsic basis $\boldsymbol{q}$ appropriately.

4.2.4 Step 3: Choice of Test Functions

It was already pointed out in section 3.1 that the choice of test functions in a weighted residual method leads to different classes of numerical methods each having its characteristic properties. In the meshbased context, one may identify popular methods like the finite element, finite volume, finite difference, and boundary element methods by choosing the test functions accordingly. The same principle holds in the meshfree context.

Fig. 4 summarizes the three steps that may be used to classify meshfree methods. Construction of a PU, choice of the approximation and test functions lead to individual names of certain MMs. For a specific MM, these three properties are in general defined, however, a few methods may occur in more than one case. For example, the partition of unity

method (PUM) may be used in a collocation or Galerkin scheme.

It may be seen that the classification in Fig. 4 is not only restricted to meshfree methods, but also includes meshbased methods. This shows important relationships of MMs to their meshbased counterparts. Furthermore, also combined meshfree/meshbased methods—as is the case for coupled PUs—can be associated with this figure. The figure shows a large number of acronyms for MMs, which will be explained in the following subsections (see also the appendix of this work). Some of the individual MMs are briefly discussed in section 4.4.

4.3 Moving Least-Squares Method

The MLS technique is the fundamental procedure of many MMs for the construction of a partition of unity of a certain order. The method was introduced by Lancaster and Salkauskas in [121] for smoothing and interpolating data.

4.3.1 Deduction of MLS functions

The method is discussed here following [121, 137]. Assuming a function $u(\boldsymbol{x})$ defined on an open set $\Omega \in \mathbb{R}^d$ is sufficiently smooth, i.e. at least $u(\boldsymbol{x}) \in C^0(\Omega)$, one can define a "local" approximation around a fixed point $\overline{\boldsymbol{x}} \in \Omega$ as

$$u^l(\boldsymbol{x}, \overline{\boldsymbol{x}}) \approx L_{\overline{\boldsymbol{x}}} u(\boldsymbol{x}) = \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x})\, \boldsymbol{a}(\overline{\boldsymbol{x}}), \tag{4.13}$$

where

$$u^l(\boldsymbol{x}, \overline{\boldsymbol{x}}) = \begin{cases} u(\boldsymbol{x}) & \forall \boldsymbol{x} \in \widetilde{\Omega}(\overline{\boldsymbol{x}}), \\ 0 & \text{otherwise.} \end{cases} \tag{4.14}$$

$\widetilde{\Omega}(\overline{\boldsymbol{x}})$ denotes a local domain around $\overline{\boldsymbol{x}}$ and the operator $L_{\overline{\boldsymbol{x}}}$ is a certain mapping. The vector $\boldsymbol{p}(\boldsymbol{x})$ is chosen according to (4.4) to be a basis of order n in dimension d; this basis is also called *intrinsic* basis. In order that the local approximation is the best approximation of u in a certain least-squares sense, the unknown coefficient vector $\boldsymbol{a}(\overline{\boldsymbol{x}})$ is selected to

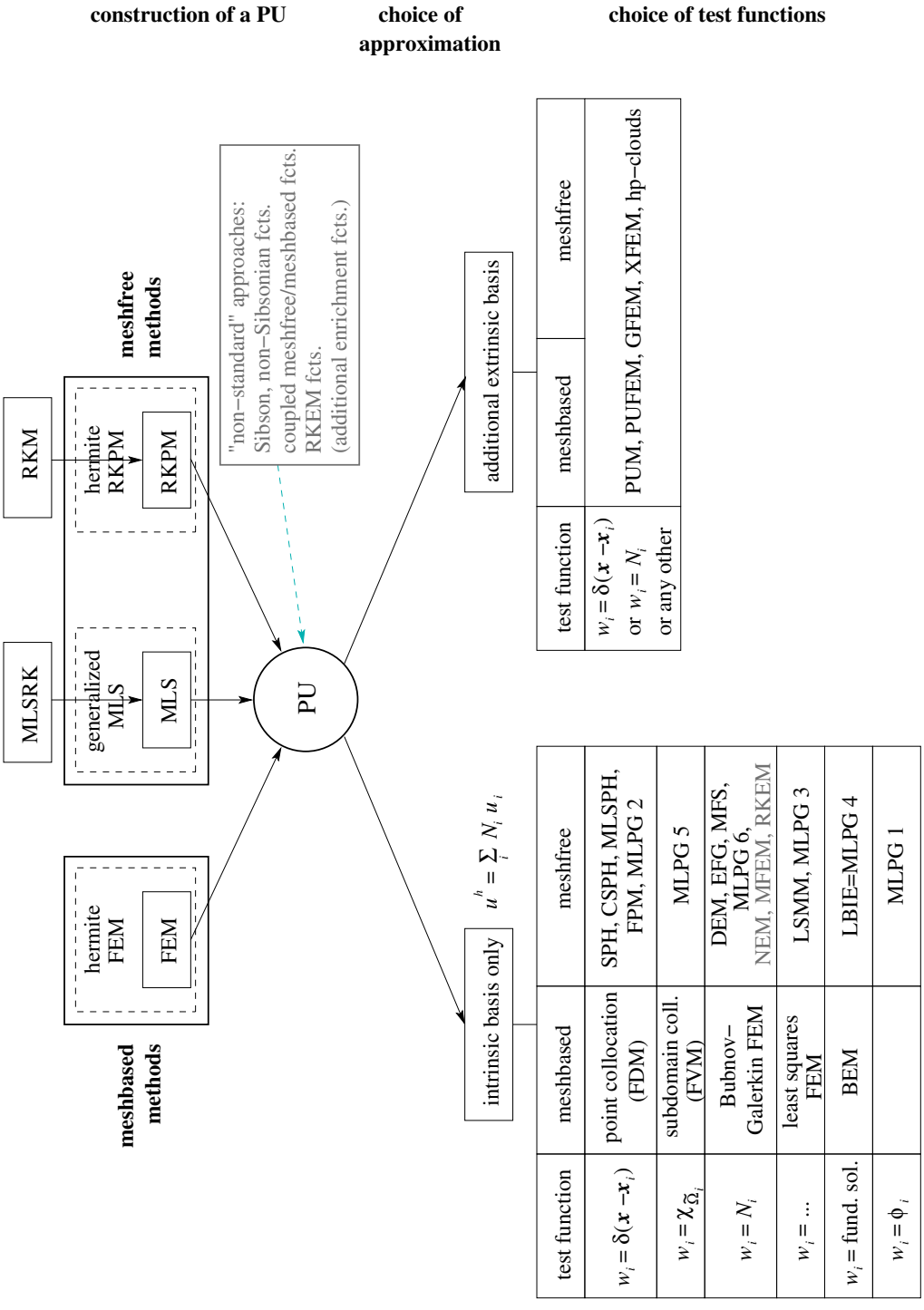

Figure 4: Classification of meshfree methods.

minimize the weighted least-squares discrete L_2 error norm

$$J_{\overline{\boldsymbol{x}}}\left(\boldsymbol{a}\left(\overline{\boldsymbol{x}}\right)\right) = \sum_{i=1}^{r} \phi\left(\overline{\boldsymbol{x}} - \boldsymbol{x}_i\right) \left[L_{\overline{\boldsymbol{x}}} u\left(\boldsymbol{x}\right) - u^l\left(\boldsymbol{x}, \overline{\boldsymbol{x}}\right)\right]^2, \tag{4.15}$$

$$= \sum_{i=1}^{r} \phi\left(\overline{\boldsymbol{x}} - \boldsymbol{x}_i\right) \left[\boldsymbol{p}^{\mathrm{T}}\left(\boldsymbol{x}_i\right) \boldsymbol{a}\left(\overline{\boldsymbol{x}}\right) - u\left(\boldsymbol{x}_i\right)\right]^2. \tag{4.16}$$

Thereby, a relation of the unknowns $\boldsymbol{a}\left(\overline{\boldsymbol{x}}\right)$ with the nodal values $\boldsymbol{u}$ is found. $\boldsymbol{x}_i$ refers to the position of the r nodes within the domain, which is discussed separately in section 4.3.2. The weighting function $\phi\left(\overline{\boldsymbol{x}} - \boldsymbol{x}_i\right)$—which is not related in any sense to the weighting (=test) functions in a weighted residual method—plays an important role in the context of MMs, which is worked out in section 4.3.3. It has small supports $\widetilde{\Omega}_i$ around each node, thereby ensuring the locality of the approximation; the overlapping situation of the supports $\widetilde{\Omega}_i$ within the domain is called cover. The weighting function may also be chosen individually for each node as $\phi_i\left(\overline{\boldsymbol{x}} - \boldsymbol{x}_i\right)$.

The functional $J_{\overline{\boldsymbol{x}}}\left(\boldsymbol{a}\left(\overline{\boldsymbol{x}}\right)\right)$ is minimized with respect to $\boldsymbol{a}\left(\overline{\boldsymbol{x}}\right)$, i.e. $\frac{\partial J}{\partial \boldsymbol{a}} = 0$, which results in the following system of k equations

$$\sum_{i=1}^{r} \phi\left(\overline{\boldsymbol{x}} - \boldsymbol{x}_i\right) \boldsymbol{p}\left(\boldsymbol{x}_i\right) \boldsymbol{p}^{\mathrm{T}}\left(\boldsymbol{x}_i\right) \boldsymbol{a}\left(\overline{\boldsymbol{x}}\right) = \sum_{i=1}^{r} \phi\left(\overline{\boldsymbol{x}} - \boldsymbol{x}_i\right) \boldsymbol{p}\left(\boldsymbol{x}_i\right) u_i. \tag{4.17}$$

Solving this for $\boldsymbol{a}\left(\overline{\boldsymbol{x}}\right)$ and then replacing $\boldsymbol{a}\left(\overline{\boldsymbol{x}}\right)$ in the local approximation (4.13) leads to

$$L_{\overline{\boldsymbol{x}}} u\left(\boldsymbol{x}\right) = \boldsymbol{p}^{\mathrm{T}}\left(\boldsymbol{x}\right) \left[\sum_{i=1}^{r} \phi\left(\overline{\boldsymbol{x}} - \boldsymbol{x}_i\right) \boldsymbol{p}\left(\boldsymbol{x}_i\right) \boldsymbol{p}^{\mathrm{T}}\left(\boldsymbol{x}_i\right)\right]^{-1} \sum_{i=1}^{r} \phi\left(\overline{\boldsymbol{x}} - \boldsymbol{x}_i\right) \boldsymbol{p}\left(\boldsymbol{x}_i\right) u_i. \tag{4.18}$$

In order to extend this local approximation to the whole domain, the so-called moving-procedure is introduced to achieve a global approximation. Since the point $\overline{\boldsymbol{x}}$ can be chosen arbitrarily, one can let it "move" over the whole domain, $\overline{\boldsymbol{x}} \to \boldsymbol{x}$, which leads to the global approximation of $u\left(\boldsymbol{x}\right)$ [137]. Mathematically a global approximation operator G is introduced with

$$u\left(\boldsymbol{x}\right) \approx G u\left(\boldsymbol{x}\right) = u^h\left(\boldsymbol{x}\right), \tag{4.19}$$

where the operator G is another mapping, defined as $G u\left(\boldsymbol{x}\right) = \lim_{\overline{\boldsymbol{x}} \to \boldsymbol{x}} L_{\overline{\boldsymbol{x}}} u\left(\boldsymbol{x}\right)$ and can be interpreted as the globalization of the local approximation operator $L_{\overline{\boldsymbol{x}}}$ through the moving process [137].

Finally, the MLS approximation may be written as

$$u^h(\boldsymbol{x}) = \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) \left[\sum_{i=1}^{r} \phi(\boldsymbol{x} - \boldsymbol{x}_i)\, \boldsymbol{p}(\boldsymbol{x}_i)\, \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}_i) \right]^{-1} \sum_{i=1}^{r} \phi(\boldsymbol{x} - \boldsymbol{x}_i)\, \boldsymbol{p}(\boldsymbol{x}_i)\, u_i, \tag{4.20}$$

or

$$u^h(\boldsymbol{x}) = \underbrace{\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x})}_{(1 \times k)} \cdot \underbrace{[\mathbf{M}(\boldsymbol{x})]^{-1}}_{(k \times k)} \cdot \underbrace{\mathbf{B}(\boldsymbol{x})}_{(k \times r)} \cdot \underbrace{\boldsymbol{u}}_{(r \times 1)}, \tag{4.21}$$

where

$$\mathbf{M}(\boldsymbol{x}) = \sum_{i=1}^{r} \phi(\boldsymbol{x} - \boldsymbol{x}_i)\, \boldsymbol{p}(\boldsymbol{x}_i)\, \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}_i) \tag{4.22}$$

and

$$\mathbf{B}(\boldsymbol{x}) = \left[\begin{array}{cccc} \phi(\boldsymbol{x} - \boldsymbol{x}_1)\, \boldsymbol{p}(\boldsymbol{x}_1) & \phi(\boldsymbol{x} - \boldsymbol{x}_2)\, \boldsymbol{p}(\boldsymbol{x}_2) & \ldots & \phi(\boldsymbol{x} - \boldsymbol{x}_r)\, \boldsymbol{p}(\boldsymbol{x}_r) \end{array} \right]. \tag{4.23}$$

The matrix $\mathbf{M}(\boldsymbol{x})$ is often called *moment matrix*; it is of size $k \times k$, i.e. of the same size than the complete basis vector $\boldsymbol{p}(\boldsymbol{x})$. This matrix has to be inverted wherever the MLS shape functions are to be evaluated. Obviously, for a higher desired order of consistency and thus higher values of k, this matrix inversion becomes more expensive.

Taking the viewpoint of an approximation of the form $u^h(\boldsymbol{x}) = \boldsymbol{N}^{\mathrm{T}}(\boldsymbol{x})\, \boldsymbol{u}$, one can immediately write for the MLS functions

$$\boldsymbol{N}^{\mathrm{T}}(\boldsymbol{x}) = \underbrace{\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) [\mathbf{M}(\boldsymbol{x})]^{-1} \mathbf{B}(\boldsymbol{x})}_{(1 \times r)}, \tag{4.24}$$

and thus for one certain shape function N_i at a point $\boldsymbol{x}$

$$N_i(\boldsymbol{x}) = \underbrace{\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) [\mathbf{M}(\boldsymbol{x})]^{-1} \phi(\boldsymbol{x} - \boldsymbol{x}_i)\, \boldsymbol{p}(\boldsymbol{x}_i)}_{(1 \times 1)}. \tag{4.25}$$

The MLS functions fulfill the consistency requirements of order n, i.e. they build a partition of unity of order n. It can easily be shown that functions of the basis $\boldsymbol{p}(\boldsymbol{x})$ are found exactly by an MLS approximation, see e.g. [21]. In practice, it is—at least for $n > 2$—almost impossible to write down the shape functions in an explicit way, i.e. without the matrix inversion. Thus, the shape functions may be evaluated at arbitrary many points, but

without knowing the shape functions explicitly. In the literature this is sometimes called "evaluating a function digitally", as it is not known in an explicit continuous ("analogous") form [4].

It is interesting to note that any linear combination of the basis functions will indeed lead to the same shape functions, see the proof e.g. in [75]. According to this, any translated and scaled basis can be used, leading to the same shape functions. This will be of importance for a better conditioning of the moment matrix, see section 4.3.4.

The first derivatives of the MLS shape functions follow according to the product rule as

$$\boldsymbol{N}^{\mathrm{T}}_{,k}(\boldsymbol{x}) = \boldsymbol{p}^{\mathrm{T}}_{,k}\mathbf{M}^{-1}\mathbf{B} + \boldsymbol{p}^{\mathrm{T}}\left(\mathbf{M}^{-1}\right)_{,k}\mathbf{B} + \boldsymbol{p}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{B}_{,k}, \tag{4.26}$$

with $\left(\mathbf{M}^{-1}\right)_{,k} = -\mathbf{M}^{-1}\mathbf{M}_{,k}\mathbf{M}^{-1}$. The second derivatives are

$$\begin{aligned}\boldsymbol{N}^{\mathrm{T}}_{,kl}(\boldsymbol{x}) \;=\;& \boldsymbol{p}^{\mathrm{T}}_{,kl}\mathbf{M}^{-1}\mathbf{B} + \boldsymbol{p}^{\mathrm{T}}_{,k}\left(\mathbf{M}^{-1}\right)_{,l}\mathbf{B} + \boldsymbol{p}^{\mathrm{T}}_{,k}\mathbf{M}^{-1}\mathbf{B}_{,l} + \\ & \boldsymbol{p}^{\mathrm{T}}_{,l}\left(\mathbf{M}^{-1}\right)_{,k}\mathbf{B} + \boldsymbol{p}^{\mathrm{T}}\left(\mathbf{M}^{-1}\right)_{,kl}\mathbf{B} + \boldsymbol{p}^{\mathrm{T}}\left(\mathbf{M}^{-1}\right)_{,k}\mathbf{B}_{,l} + \\ & \boldsymbol{p}^{\mathrm{T}}_{,l}\mathbf{M}^{-1}\mathbf{B}_{,k} + \boldsymbol{p}^{\mathrm{T}}\left(\mathbf{M}^{-1}\right)_{,l}\mathbf{B}_{,k} + \boldsymbol{p}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{B}_{,kl},\end{aligned} \tag{4.27}$$

with $\left(\mathbf{M}^{-1}\right)_{,kl} = \mathbf{M}^{-1}\mathbf{M}_{,l}\mathbf{M}^{-1}\mathbf{M}_{,k}\mathbf{M}^{-1} - \mathbf{M}^{-1}\mathbf{M}_{,kl}\mathbf{M}^{-1} + \mathbf{M}^{-1}\mathbf{M}_{,k}\mathbf{M}^{-1}\mathbf{M}_{,l}\mathbf{M}^{-1}$. In [20], Belytschko *et al.* propose an efficient way to compute the derivatives of the MLS shape functions by means of a LU decomposition of the $k \times k$ system of equations.

As an example, Fig. 5 shows MLS shape functions with first order consistency and their derivatives in a one-dimensional domain $\Omega = (0, 1)$ with 11 equally distributed nodes. The weighting functions—discussed in detail in section 4.3.3—have a dilatation parameter of $\rho = 3 \cdot \Delta x = 0.3$. The following important properties can be seen:

- The dashed line in the upper picture shows that the sum of the shape functions $\sum_i N_i(x)$ equals 1 in the whole domain, thus $\{N_i\}$ builds a PU. The derivatives of the MLS-PU build "partition of nullities", i.e. $\sum_i \partial_x N_i(x) = \sum_i \partial_{xx} N_i(x) = 0$.

- The non-polynomial shape functions themselves are smooth and can still be regarded to be rather polynomial-like, but the derivatives tend to have a more and more non-polynomial character. This causes problems in integrating the integral expressions of the weak form, see section 4.3.6.

- The shape functions are not interpolating, i.e. they do not possess the Kronecker-δ property. The unknowns $\boldsymbol{u}$ of a meshfree approximation are *not* nodal values. Due

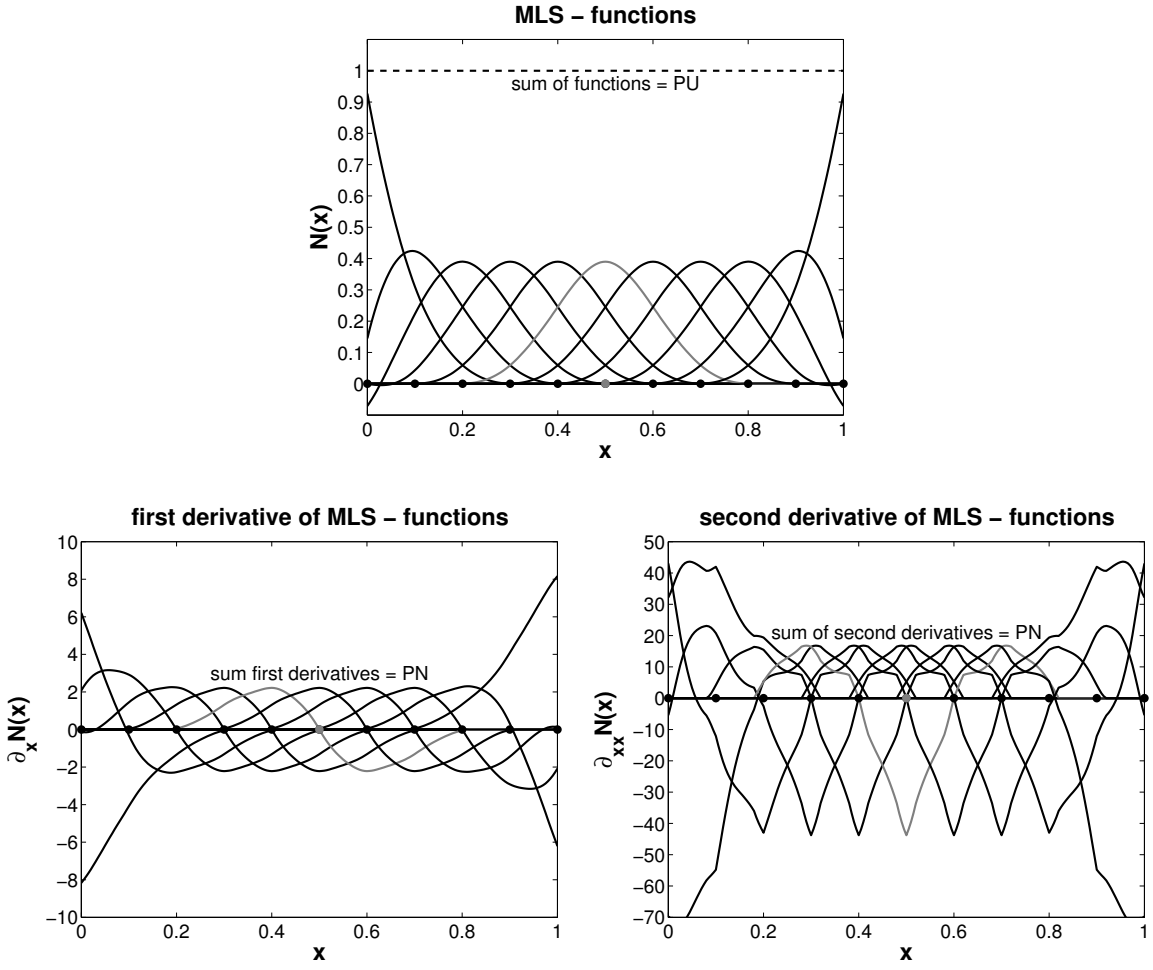

Figure 5: Partition of unity functions and derivatives constructed with the MLS technique.

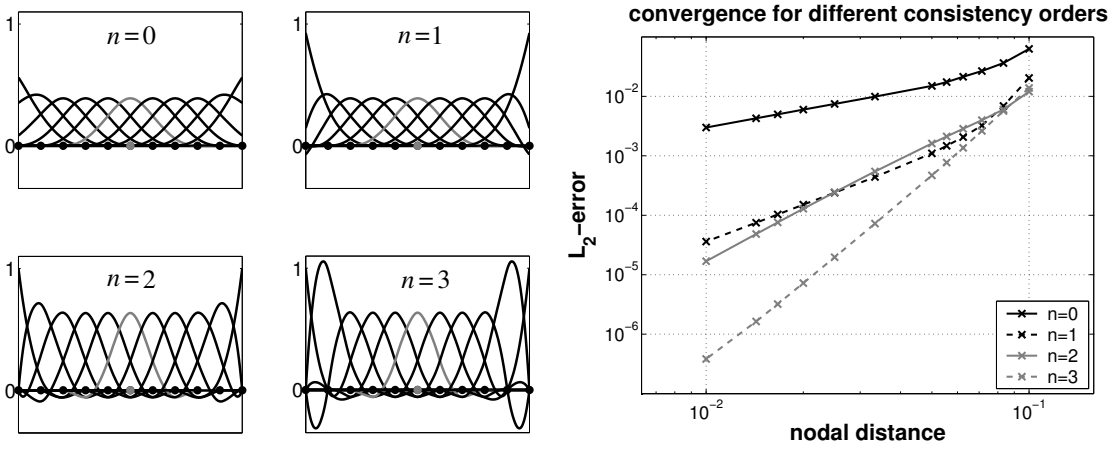

Figure 6: The left part show PUs of different order n, the right part a convergence analysis of an example problem solved with the different PUs.

to this fact, they are sometimes called fictitious values [4]. To have the real values of the sought function $u(\boldsymbol{x})$ at a point, all influences of shape functions which are non-zero there have to be added up. The non-interpolant character makes imposition of essential boundary conditions difficult, see section 4.3.5.

Fig. 6 shows PUs in a one-dimensional domain with different consistency orders. These functions have been used in a Bubnov-Galerkin setting for the approximation of the one-dimensional advection-diffusion equation $c \cdot u_x - K \cdot u_{,xx} = f$ in $\Omega = (0,1)$ with $u(0) = 0$, $u(1) = 1$. The right hand side is prescribed such that the exact solution is $u_{\text{ex}} = \sin(5/2 \cdot \pi \cdot x)$. A convergence analysis for different node numbers is shown in the right part of the figure. One may see in this case that the consistency order of the shape functions is directly correlated to the order of convergence of the resulting method.

Different Approaches for the Deduction of the MLS. The same result for the MLS functions may be obtained in several ways. One may start with an ansatz of the form $N_i(\boldsymbol{x}) = \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x})\,\boldsymbol{a}(\boldsymbol{x})\,\phi(\boldsymbol{x} - \boldsymbol{x}_i)$. A possible approach is based on a Taylor series expansion

$$u_i = u(\boldsymbol{x}_i) = \sum_{|\boldsymbol{\alpha}|=0}^{\infty} \frac{(\boldsymbol{x}_i - \boldsymbol{x})^{\boldsymbol{\alpha}}}{|\boldsymbol{\alpha}|!} D^{\boldsymbol{\alpha}} u(\boldsymbol{x}), \tag{4.28}$$

which is inserted into $u^h(\boldsymbol{x}) = \sum_i N_i(\boldsymbol{x})\,u_i$. Then, $\boldsymbol{a}$ is determined such that the first k terms in the Taylor series are exactly reproduced; the n-th order consistency of the MLS

functions is then obvious. Another approach is to directly insert the ansatz for $N_i(\boldsymbol{x})$ into the consistency conditions (4.1). Both approaches are worked out in [56] by Fries and Matthies, see also [21, 32].

Equivalence to the reproducing kernel particle method (RKPM). The MLS works from the beginning with a *discrete* set of r nodes distributed in a domain Ω. In contrast, one may develop a *continuous* background for the MLS. This background is called moving least-squares reproducing kernel (MLSRK) [137] and provides a link to the reproducing kernel (particle) methods [133, 136, 134]. A reproducing kernel method (RKM) is a *continuous* approximation of the kind

$$u^h(\boldsymbol{x}) = \int_{\Omega_{\boldsymbol{y}}} K(\boldsymbol{x}, \boldsymbol{y})\, u(\boldsymbol{y})\, d\Omega, \tag{4.29}$$

where $K(\boldsymbol{x}, \boldsymbol{y})$ is called kernel. Clearly, if $K(\boldsymbol{x}, \boldsymbol{y})$ equals the Dirac function $\delta(\boldsymbol{x}, \boldsymbol{y})$, the function $u(\boldsymbol{x})$ will be reproduced exactly. For the evaluation of such an integral in practice, the RKM has to be discretized, hence

$$u^h(\boldsymbol{x}) = \sum_{i=1}^{r} K(\boldsymbol{x}_i - \boldsymbol{x}, \boldsymbol{x})\, u_i \Delta V_i, \tag{4.30}$$

where ΔV_i are suitable integration weights. This *discrete* version is called reproducing kernel particle method (RKPM). The kernel of the RKM, (4.29), may be constructed such that this approximation is able to reproduce polynomials of a certain order exactly:

$$u^h(\boldsymbol{x}) = \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) \left[\int_{\Omega_{\boldsymbol{y}}} \phi(\boldsymbol{x} - \boldsymbol{y})\, \boldsymbol{p}(\boldsymbol{y})\, \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{y})\, d\Omega \right]^{-1} \int_{\Omega_{\boldsymbol{y}}} \phi(\boldsymbol{x} - \boldsymbol{y})\, \boldsymbol{p}(\boldsymbol{y})\, u(\boldsymbol{y})\, d\Omega. \tag{4.31}$$

Discretization of these continuous expression leads to the RKPM which is equivalent to (4.20). However, a slight difference may be mentioned: The integration weights emerging from the discretization of (4.31) are 1 for the MLS [137], but may be chosen differently in the RKPM. However, in practice one does not often take advantage of this additional freedom, for an exception see [2].

The equivalence between MLS and RKPM is a remarkable result which unifies two methodologies with very different origins, it has also been discussed in [21, 125, 123, 133]. Belytschko *et al.* claim in [21]:

Any kernel method in which the parent kernel is identical to the weight function of a MLS approximation and is rendered consistent by the same basis is identical. In other words, a discrete kernel approximation which is consistent must be identical to the related MLS approximation.

Generalized moving least-squares. It is possible to treat the derivatives of a function as independent functions, this is the idea of the generalized MLS [3]. This can for example be important for the solution of forth order boundary value problems (e.g. analysis of thin beams), where displacement and slope boundary conditions might be imposed at the same point. The local approximation (4.13) remains unchanged, however, the following weighted discrete H^l error norm is used instead of the above used L_2 error norm (4.16):

$$J_{\overline{\boldsymbol{x}}}^{(m)}\left(\boldsymbol{a}\left(\overline{\boldsymbol{x}}\right)\right)=\sum_{j=1}^{m}\sum_{i=1}^{r}\phi^{\left(\boldsymbol{\alpha}^j\right)}\left(\overline{\boldsymbol{x}}-\boldsymbol{x}_i\right)\left[D^{\boldsymbol{\alpha}^j}\boldsymbol{p}^{\mathrm{T}}\left(\boldsymbol{x}_i\right)\boldsymbol{a}\left(\overline{\boldsymbol{x}}\right)-D^{\boldsymbol{\alpha}^j}u\left(\boldsymbol{x}_i\right)\right]^2, \tag{4.32}$$

with $\phi^{\left(\boldsymbol{\alpha}^j\right)}=D^{\boldsymbol{\alpha}^j}\phi$. Minimization of this functional still leads to a system of k equations, the extra effort lies in building m times the sum over all points, with $m\leq k$ being the number of derivatives which shall be included in the approximation as unknowns [3]. Without repeating the details of the moving procedure, the approximation results in

$$\begin{aligned} u^h\left(\boldsymbol{x}\right) &= \sum_{j=1}^{m}\left(\boldsymbol{p}^{\mathrm{T}}\left(\boldsymbol{x}\right)\left[\sum_{i=1}^{r}\phi^{\left(\boldsymbol{\alpha}^j\right)}\left(\boldsymbol{x}-\boldsymbol{x}_i\right)D^{\boldsymbol{\alpha}^j}\boldsymbol{p}\left(\boldsymbol{x}_i\right)D^{\boldsymbol{\alpha}^j}\boldsymbol{p}^{\mathrm{T}}\left(\boldsymbol{x}_i\right)\right]^{-1}\right. && (4.33)\\ &\quad \left.\sum_{i=1}^{r}\phi^{\left(\boldsymbol{\alpha}^j\right)}\left(\boldsymbol{x}-\boldsymbol{x}_i\right)D^{\boldsymbol{\alpha}^j}\boldsymbol{p}\left(\boldsymbol{x}_i\right)D^{\boldsymbol{\alpha}^j}u_i\right). && (4.34)\end{aligned}$$

The same idea has been proposed in the RKPM context under the name "Hermite RKPM" [134]. This name stems from the well-known Hermite (or Hermitian) polynomials used in the FEM for the solution of forth order boundary value problems [191].

Shepard's method. It can be easily shown that (4.20) leads to zero-order consistency, i.e. a basis of $\boldsymbol{p}\left(\boldsymbol{x}\right)=1$, to the Shepard interpolants [169] defined as

$$u^h\left(\boldsymbol{x}\right)=\sum_{i=1}^{r}N_i\left(\boldsymbol{x}\right)u_i,\quad \text{with}\quad N_i\left(\boldsymbol{x}\right)=\frac{\phi\left(\boldsymbol{x}-\boldsymbol{x}_i\right)}{\sum_{i=1}^{r}\phi\left(\boldsymbol{x}-\boldsymbol{x}_i\right)}. \tag{4.35}$$

These functions form a particularly simple and efficient way to find a meshfree PU of 0-th order. The problem is clearly the low order of consistency which make the Shepard PU fail for the solution of even second order boundary value problems. But, it shall be mentioned that ways have been shown to construct a linear-precision (=first order consistent) PU based on the Shepard's method with only small extra computational effort [117]. Furthermore, it is mentioned in [21] that in fact Shepard functions have been used for the simulation of second-order PDEs, showing that consistency is sufficient (stability provided) but may not be necessary for convergence.

Relation to other least-squares schemes. Oñate *et al.* point out in [159] that any least-squares scheme can be used for an approximation, hence for obtaining certain shape functions. The basic idea is always to minimize

$$J = \sum_{i=1}^{r} \phi \left(u^h \left(\boldsymbol{x}_i \right) - u \left(\boldsymbol{x}_i \right) \right)^2 = \sum_{i=1}^{r} \phi \left(\boldsymbol{p}^{\mathrm{T}} \left(\boldsymbol{x}_i \right) \boldsymbol{a} - u_i \right)^2, \tag{4.36}$$

where different choices for the weighting function ϕ lead to different lest-squares schemes. In particular, the choice $\phi = 1$ leads to the standard least-squares method with the major drawback that all nodal values show influence in the whole domain. Fixed least-squares are obtained for $\phi = \phi_j \left(\boldsymbol{x}_i \right)$, where the weighting functions have their highest value at the nodal positions. ϕ_j might not be the only function which is non-zero at $\boldsymbol{x}$, and the choice of the "belonging" weighting function is not always easy; but more than one weighting function would clearly lead to a multivalued interpolation. The latter drawback is eliminated by the multiple fixed least-squares method, resulting for $\phi = \phi_i \left(\boldsymbol{x} \right)$. The MLS is obtained for $\phi = \phi \left(\boldsymbol{x} - \boldsymbol{x}_i \right)$. There, the weighting function ϕ is defined in shape and size and is translated over the domain so that it takes the maximum value at an arbitrary point $\boldsymbol{x}$, where the unknown of the function u is to be evaluated; the parameters $\boldsymbol{a}$ result as functions of $\boldsymbol{x}$.

The relations of these least-squares schemes are depicted in Fig. 7. Any of these schemes can be used for the construction of meshfree shape functions. However, in practical use the MLS shape functions are most often chosen and have therefore been presented in detail. It should also been mentioned that it is possible to obtain the conventional meshbased finite element shape functions via a least-squares procedure if the weighting function ϕ depends on a mesh, see e.g. [154].

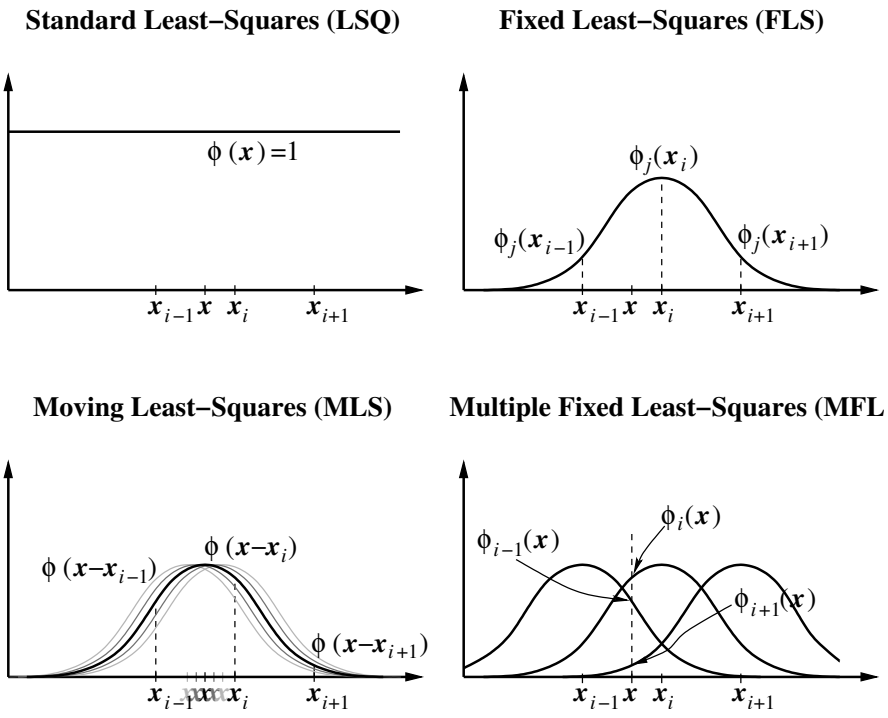

Figure 7: Different least-squares schemes.

4.3.2 Particle Placement

Starting point for any construction of a PU is the distribution of nodes in the domain. Although it is often stated that MMs work with randomly or arbitrary scattered points, the method cannot be expected to give suitable results if several criteria are not fulfilled.

There are methods for producing well-spaced point sets, similar to mesh generators for meshbased methods. Some methods rely on advancing front methods, such as the biting method [129, 128]. Other point set generators are octree based [111] or they use Voronoi diagrams and weighted bubble packing [41]. This is not considered in more detail because there are basically the same methods for the distributions of nodes as in meshbased methods where this also forms the first step.

4.3.3 Weighting Functions

The MLS procedure—as well as the RKPM—employ a weighting (=kernel or window) function ϕ, which requires further specification. These functions ensure the locality of

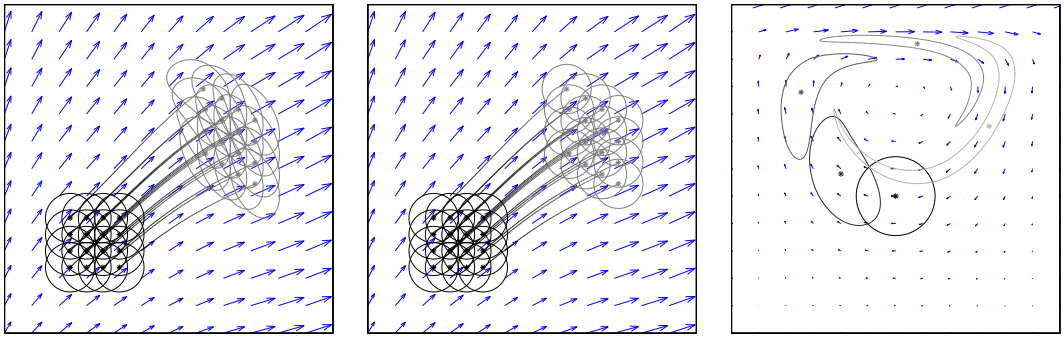

Figure 8: a) and b) compare Lagrangian and Eulerian kernels, c) shows the limited use of Lagrangian kernels. The initial situation at $t = 0$ is plotted in black, grey lines show situations at $t > 0$.

the point data due to the compact support of the resulting MLS functions. The most important characteristics of weighting functions are listed in the following.

Lagrangian and Eulerian kernels. In MMs, the particles (=nodes with meshfree shape functions) often move through the domain with certain velocities. That is, the problem under consideration is given in Lagrangian formulation, rather than in Eulerian form where particles are kept fixed throughout the calculation. Also the weighting function may be a function of the material or Lagrangian coordinates $\boldsymbol{X}$, $\phi_i(\boldsymbol{X}) = \phi(\|\boldsymbol{X} - \boldsymbol{X}_i\|, \rho)$, or of the spatial or Eulerian coordinates $\boldsymbol{x}$, $\phi_i(\boldsymbol{x}) = \phi(\|\boldsymbol{x} - \boldsymbol{x}_i(t)\|, \rho)$. The difference between these two formulations may be seen in Fig. 8a) and b), where particles move due to a prescribed non-divergence-free velocity field. It is obvious that the shape of the support changes with time for the Lagrangian kernel but remains constant for the Eulerian kernel.

An important consequence of the Lagrangian kernel is that neighbours of a particle remain neighbours throughout the simulation. This has large computational benefits, because a neighbour search for the summation of the MLS system of equations has only to be done once at the beginning of a computation. In addition, it has been shown in [17] and [163] that Lagrangian kernels have superior stability properties in Lagrangian collocation MMs. However, the usage of Lagrangian kernels comes at the major disadvantage that it is limited to computations with rather small movements of the particles during the calculation (as is often the case e.g. in structural mechanics) [163]. It can be seen in Fig. 8c) that Lagrangian kernels may not be used in problems of fluid dynamics due to the prohibitive large deformation of the support shape. In this figure a divergence-free flow field

resulting from the well-known driven cavity test case has been taken as an example. Clearly, in cases where neighbour relations break naturally—i.e. physically justified—throughout the simulation, Lagrangian kernels seem useless as their property to keep the neighbour relations constant is undesirable.

It is clear that if the problem under consideration is posed in an Eulerian form, then the particle positions are fixed throughout the simulation and the shape of the supports stays constant, i.e. Eulerian kernels result naturally. For a theoretical analysis of Lagrangian and Eulerian kernels see [17].

Size and shape of the support. The support $\widetilde{\Omega}_i$ of a weighting function ϕ_i differs in size and shape, the latter including implicitly the dimension of the PDE problem under consideration. Although any choice of the support shape might be possible, in practice spheres, ellipsoids and parallelepipeds are most frequently used. The size and shape of the support of the weight function is directly related to the size and support of the resulting shape function, and $N_i(\boldsymbol{x}) = 0 \;\forall\; \{\boldsymbol{x} \,|\phi_i(\boldsymbol{x}) = 0\}$.

The size of the support is defined by the so-called dilatation parameter or smoothing length ρ. It is critical to solution accuracy and stability and plays a role similar to the element size in the FEM. h-refinement in the FEM can be produced in MMs by decreasing the value of the dilatation parameter, thus implying an increase in the density of the particles [136]. Although the dilatation parameter is often chosen to be constant for all points $\boldsymbol{x}_i$ it can be different for each point and may vary during the calculation. The aim is to obtain "good" solutions—although here it remains unclear how to find optimal smoothing lengths [136]—or to keep the number of particles in the support of each node constant [85]. In both cases, one needs to determine time derivatives of ρ, leading to complicated equations. However, Gingold and Monoghan found in [68] that if these terms are omitted, energy is conserved with an error $< 1\%$ or less for large particle numbers r.

Any one-dimensional weighting function $\phi(x)$ can be used to create a d-dimensional weighting function either of the form $\phi(\|\boldsymbol{x}\|)$ in case of spherical supports or by a tensor product $\prod_{i=1}^{d} \phi(x_i)$ in case of parallelepipeds.

The intersecting situation of supports $\widetilde{\Omega}_i$ is also called cover [71]. The cover construction, i.e. the choice of the size (implicitly through the dilatation parameter ρ) and shape of the supports has to fulfill—together with the node distribution—certain conditions in order to ensure the regularity of the $k \times k$ system of equations (moment matrix) which arises

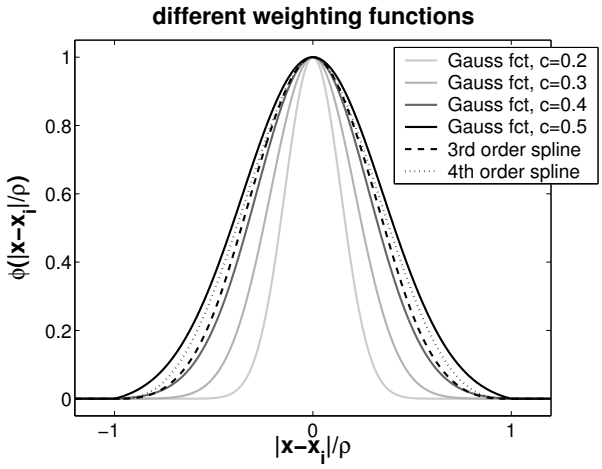

Figure 9: Exponential and spline weighting functions.

in the MLS procedure, see section 4.3.4. The aspect of an automatic cover construction for a given point set is worked out in [71, 78]. However in practice, instead of using certain algorithms for the definition of the cover, it is often constructed manually in a straightforward "intuitive" way.

Functional form of the weighting function. The third important characteristic of weight functions is their *functional form*. In general, ϕ is chosen to be a member of a sequence of functions which approximates the Dirac-δ function [68], motivated by (4.29). There exist infinitely many possible choices [76, 149] but typically, bell-shaped (Gaussian-like) functions are used. The functional form has some effect on the convergence of an approximation, which is difficult to predict [136].

An important consequence of the choice of the functional form is the continuity (smoothness) of the approximation. The smoothness of the resulting shape function is directly related to the smoothness of the weight function. Provided that the basis $\boldsymbol{p}$ is also at least as continuous as the weighting function ϕ, then if ϕ is continuous together with its first l derivatives, i.e. $\phi \in C^l(\Omega)$, the interpolation is also continuous together with its first l derivatives. More general, if $\boldsymbol{p} \in C^m(\Omega)$ and $\phi \in C^l(\Omega)$, then the shape function $\boldsymbol{N} \in C^{\min(l,m)}(\Omega)$, see e.g. [47] for a proof.

Some examples of frequently used weighting functions are:

$$
\begin{aligned}
3^{rd}\text{ order spline}: \quad \phi(q) \in C^2 &= \begin{cases} \frac{2}{3} - 4q^2 + 4q^3 & q \le \frac{1}{2} \\ \frac{4}{3} - 4q + 4q^2 - \frac{4}{3}q^3 & \frac{1}{2} < q \le 1 \\ 0 & q > 1 \end{cases}, \\
4^{th}\text{ order spline}: \quad \phi(q) \in C^2 &= \begin{cases} 1 - 6q^2 + 8q^3 - 3q^4 & q \le 1 \\ 0 & q > 1 \end{cases}, \\
2k^{th}\text{ order spline}: \quad \phi(q) \in C^{k-1} &= \begin{cases} (1-q^2)^k & q \le 1 \\ 0 & q > 1 \end{cases}, \\
\text{singular}: \quad \phi(q) \in C^0 &= \begin{cases} q^{-k} - 1 & q \le 1 \\ 0 & q > 1 \end{cases}, \\
\text{exponential 1}: \quad \phi(q) \in C^{-1} &= \begin{cases} e^{-(q/c)^{2k}} & q \le 1 \\ 0 & q > 1 \end{cases}, \\
\text{exponential 2}: \quad \phi(q) \in C^0 &= \begin{cases} \frac{e^{-(q/c)^{2k}} - e^{-(1/c)^{2k}}}{1 - e^{-(1/c)^{2k}}} & q \le 1 \\ 0 & q > 1 \end{cases}, \\
\text{exponential 3}: \quad \phi(q) \in C^\infty &= \begin{cases} e^{1/\left(q^2-1\right)} & q \le 1 \\ 0 & q > 1 \end{cases},
\end{aligned}
\tag{4.37}
$$

where $q = \|\boldsymbol{x} - \boldsymbol{x}_i\| / \rho$. In Fig. 9, the third and forth order spline weighting functions are shown together with the exponential (Gaussian) weighting function (version 2) for different values of c and $k = 1$.

As discussed in section 4.3.1, the MLS approximant $u^h = Gu$, does in general not interpolate ("pass through") the data, which might be disadvantageous. Already Lancaster and Salkauskas pointed out in [121] that the interpolating (Kronecker-δ) property of the shape functions can be recovered by using singular weighting functions at all nodes.

It is mentioned that the support size (defined by ρ) and shape as well as the functional form of the weighting function are free values. It is impossible to choose these values in a general, optimal way suited for arbitrary problems under consideration. However, one may take advantage of these free values in obtaining certain properties of the approximation method. For example, in [106] Jin *et al.* modify the weighting in the framework of meshfree collocation methods in order to enable the fulfillment of the so-called positivity conditions (which also arise in a finite difference context). Atluri *et al.* modify either the functional form of the weighting function or shift the support in upwind direction in order to obtain stabilizing effects in a Galerkin setting for advection-dominated problems [4]. On the other

hand, it may also be a major problem in certain cases to have free values to define the characteristics of the weighting function without knowing how to choose them. For example, intuitive *ad hoc* approaches such as keeping the ratio between the particle density and smoothing length ρ/h constant when changing the particle number locally seems straightforward, however, in the context of standard SPH this may not even converge [19]. Or an improper choice of a parameter may result in instability of the numerical solution (small changes of improperly selected parameter evoke large fluctuations in the solutions), see e.g. [5]. In these situations it is obviously not desirable to have these free values. Despite of these considerations, it should be added that it is in practice often not difficult to choose the free parameters and obtain satisfactory results.

4.3.4 Solving the $k \times k$ System of Equations

In the MLS method (and RKPM)—in order to evaluate the n-th order consistent shape functions at a certain point $\boldsymbol{x}$—a $k \times k$ matrix, the moment matrix $\mathbf{M}(\boldsymbol{x})$, see (4.22), must be inverted, i.e. a system of equations must be solved. The parameter k, which defines the size of this system, equals the number of components in the intrinsic basis $\boldsymbol{p}(\boldsymbol{x})$, and thus depends on the dimension of the problem d and the desired consistency order n, see section 4.2.1. *The need to build up and invert the moment matrix at a large number of integration points is the major drawback of MMs*, because of the computational cost and the possibility that the matrix inversion fails. The computational cost consists in evaluating summation expressions including a neighbour search and in matrix inversion itself. Furthermore, the computation of the derivatives of the shape functions involves a large number of (small) matrix-matrix and matrix-vector multiplications, see (4.26) and (4.27).

Evaluating the summation expressions in (4.22) and (4.23) requires the identification of the particles' neighbours, i.e. the detection of particles with $\phi(\boldsymbol{x} - \boldsymbol{x}_i) \neq 0$. This may be called connectivity computation; it is important to note that in meshbased methods, the mesh defines the connectivity *a priori*. In MMs the connectivity is determined at run-time for each point at which the shape functions need to be evaluated. This important step can dominate the total CPU time for large node numbers, especially if sequential searches, which are of $O(r)$ complexity, are used for each evaluation point. Therefore, one may use search techniques which employ localization, since such techniques can perform the search at a given point in an optimal time $O(\log r)$ [118].

The moment matrix $\mathbf{M}(\boldsymbol{x})$ is symmetric and under certain conditions it is expected to be

positive-definite. The matrix inversion is usually done via a factorization by the pivoting LU, QR factorization or singular value decomposition (the latter two are indicated for ill-conditioned matrices) [118]. A factorization of $\mathbf{M}(\boldsymbol{x})$ can be reused for the calculation of the shape function derivatives [20].

In addition to the disadvantage of the high computational burden associated with the construction and inversion of the matrix, the inversion can even fail if $\mathbf{M}(\boldsymbol{x})$ becomes singular (the rank of $\mathbf{M}(\boldsymbol{x})$ becomes smaller than k) or "nearly" singular, hence ill-conditioned. Conditions on the particle distribution (section 4.3.2) and cover (section 4.3.3) in order to ensure the regularity of the mass matrix are:

- For every point $\overline{\boldsymbol{x}} \in \Omega$ there exists a ball $B(\boldsymbol{x}) = \left\{\boldsymbol{x} \middle| \, |\boldsymbol{x} - \overline{\boldsymbol{x}}| \leq c\right\}$ in which the number of particles $r^\star$ satisfies the condition $0 < r_{\min} \leq r^\star \leq r_{\max} < \infty$ where $r_{\min}$ and $r_{\max}$ are *a priori* numbers [137].
- Each particle at position $\boldsymbol{x}_i$ has a corresponding support $\widetilde{\Omega}_i$ (where $\phi(\boldsymbol{x} - \boldsymbol{x}_i) \neq 0$). The union of all supports covers the whole domain, i.e. $\Omega \subseteq \bigcup_{i=1}^{r} \widetilde{\Omega}_i$ [137].
- Every point $\boldsymbol{x} \in \Omega$ must lie in the area of influence of at least $k = \dim(\mathbf{M})$ particles [86], hence:

$$\operatorname{card}\left\{\boldsymbol{x}_i \middle| \, \phi(\boldsymbol{x} - \boldsymbol{x}_i) \neq 0\,,\; i \in \{1, \ldots, r\}\right\} \geq k = \dim(\mathbf{M})\,. \tag{4.38}$$

- The particle distribution must be non-degenerate [86, 137]. For example, $d + 1$ particles are needed for the construction of a PU of *first* order and they must describe a non-degenerate d-simplex: In two dimensions $\boldsymbol{x}$ must belong to at least three supports of particles which are not aligned, and in three dimensions $\boldsymbol{x}$ must belong to at least four supports of particles which are not coplanar.

A robust algorithm should always check the success of the matrix inversions [118]. One way is to estimate the condition number of $\mathbf{M}(\boldsymbol{x})$, and another one ensures the final accuracy of the shape functions by checking the fulfillment of the consistency conditions (4.1), possibly including the derivative consistency conditions (4.7). Thus it is checked if really a PU of the desired order has been obtained. If a certain mismatch is exceeded an error exit can be made. Of course, satisfying the conditions for regular matrices $\mathbf{M}$ does not ensure the regularity (and consequently solvability) of the global system of equations resulting from the integration of the weak form under consideration [86].

In the MLS the basis of monomials is usually used for $\boldsymbol{p}(\boldsymbol{x})$, and this can easily lead to ill-conditioned moment matrices. However, it was already pointed out in section 4.3.1 that any linear combination of the basis functions will lead to the same shape functions. According to this, translated and scaled bases can be used leading to a better conditioning of the moment matrices. In practice, often $\boldsymbol{p}((\boldsymbol{x}-\boldsymbol{x}_i)/\rho)$ is taken instead of $\boldsymbol{p}(\boldsymbol{x})$ [86].

4.3.5 Imposing Essential Boundary Conditions

Due to the lack of Kronecker-δ property of most of the meshfree shape functions the imposition of essential boundary conditions (EBCs) requires certain attention. A number of techniques have been developed to perform this task. It seems that the imposition of EBCs in MMs is only a solved problem in the sense that it is easily possible to fulfill the prescribed boundary values directly at the nodes. However, as e.g. noted in [77], a degradation in the convergence order may be found for most of the imposition techniques in two or more dimensions for consistency orders higher than 1.

Following [56], one may divide the various methods in those that modify the weak form, those that employ shape functions with Kronecker-δ property along the essential boundary and others. These methods are briefly described in the following, together with references where this method has been used in the context of MMs.

Approaches modifying the weak form.

- **Lagrangian multiplier method** [21]: This method is a very general and accurate approach. A problem is that Lagrangian multipliers need to be solved in addition to the discrete field variables, and a separate set of interpolation functions for Lagrangian multipliers is required and must be chosen carefully with respect to the Babuška-Brezzi stability condition [30]. Furthermore, the global matrix is not positive definite and possesses zeros on its main diagonal, this restricts the choice of possible solvers [22, 141].

- **Penalty approach** [156]: In this case a penalty term of

$$\alpha \int \boldsymbol{w}\,(u-\hat{u})\,d\Gamma, \tag{4.39}$$

with $\alpha \gg 1$ is added to the weak form of the problem, and $\hat{u}$ is the prescribed value along the Dirichlet boundary. The success of this method is directly related to the usage of large numbers for α. This on the contrary influences the condition number of the resulting system of equations in a negative way, i.e. the system is more and more ill-conditioned with increasing values for α. The advantages of the penalty approach is that the size of the system of equations is constant and the possible positive definiteness remains for large enough α.

- **Nitsche's method** [8, 51]: This method may be considered a consistent improvement of the penalty method. That is, rather than adding only one term to the weak form of the problem a number of terms is added depending on the specific problem under consideration. The α-value may be chosen considerably smaller than in the Penalty method—avoiding an ill-conditioning—, and the advantages of the Penalty method remain.

Approaches employing shape functions with Kronecker-δ property.

- **Coupling with finite elements** [33, 88, 113, 185]: Any of the coupling methods to be discussed in section 6 may be used to employ a string of elements along the essential boundaries and to combine the FE shape functions defined on this string with the meshfree approximation, see Fig. 10. The advantage of this approach is clearly that all shape functions related to the essential boundary have Kronecker-δ property as they are standard FEM functions and the essential boundary conditions may be easily imposed. The disadvantage is that a string of elements has to be generated, and that the coupling necessarily leads to a somewhat complicated code structure.

- **Transformation method** [32, 36, 125]: There exist a full and a partial (boundary) transformation method. In the first, an inversion of a $r \times r$ matrix is required and the Kronecker-δ property is obtained at all nodes, whereas in the latter, only a reduced system has to be inverted and the Kronecker-δ property is obtained at boundary nodes only. The full transformation method is described as follows. Assume that $\mathbf{A}(\boldsymbol{u}) = \boldsymbol{b}$ is the general system of equations to be solved is, where $\boldsymbol{u}$ are the unknowns for a non-interpolating approximation. Then, $\mathbf{D}^{-1}\boldsymbol{u}$ with $D_{ij} = N_i(\boldsymbol{x}_j)$ are the interpolating "real" nodal values and $\mathbf{A}^\star(\mathbf{D}^{-1}\boldsymbol{u}) = \boldsymbol{b}^\star$ is solved instead. In this system of equations, the EBCs may be directly inserted.

- **Singular weighting functions** [36, 109, 121]: Using weighting functions in the MLS procedure, which are singular at the corresponding particle position, recovers the Kronecker-δ property of the MLS functions. One may also employ singular weighting functions only for the boundary nodes, then the Kronecker-δ property is only recovered at these nodes. Integration points have to be placed carefully in this approach, considering a minimum distance from the singularities. Accuracy is often a problem of this approach [79].

- **Extended approximation spaces** [8, 21, 125]: Referring to section 4.4.3, the essential boundary conditions can be implemented by choosing the local approximation spaces such that the functions satisfy the Dirichlet boundary conditions [7, 8]. For example in [21, 125] Legendre polynomials are used as an extrinsic basis recovering the Kronecker-δ property.

Others.

- **Boundary collocation** [151, 184, 189]: This method is a simple extension of the standard FEM technique for imposing essential boundary conditions. The boundary condition $u = g$ along the Dirichlet boundary Γ_{D} is enforced by $u(\boldsymbol{x}_i) = \boldsymbol{N}^{\mathrm{T}}(\boldsymbol{x}_i)\,\boldsymbol{u} = g(\boldsymbol{x}_i)$ [151]. This expression is taken directly as one equation in the total system of equations. This method can enforce EBCs exactly only at boundary points, but not in between these nodes [4]. It should be noted that an important condition is not fulfilled for the standard boundary collocation method: It is required that the test functions in a weak form must vanish along the essential boundary [184]. Neglecting this leads to a degradation in the convergence order for meshfree shape functions with high consistency orders. Therefore, Wagner and Liu propose a corrected collocation method in [184], which considers the problem of non-vanishing test functions along the essential boundary. This idea is further considered and modified in [189].

4.3.6 Integration

The integral expressions of the approximated weak form of a PDE—emanating from the weighted residual method—have to be evaluated numerically. As has been mentioned previously, this is the most time-consuming part in MMs, because of the large number of

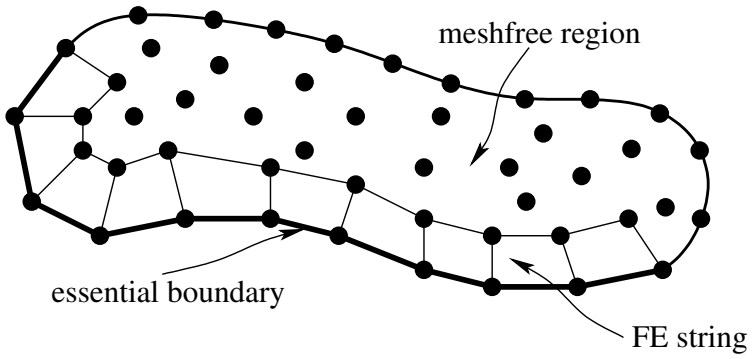

Figure 10: Using a finite element string along the essential boundary for imposition of EBCs.

integration points needed, and the computational effort in evaluating the meshfree shape functions at each integration point.

Numerical integration rules are of the form

$$\int f(\boldsymbol{x})\, d\Omega = \sum_i f(\boldsymbol{x}_i)\, \Delta V_i, \tag{4.40}$$

and vary only with regard to the locations $\boldsymbol{x}_i$ and weights ΔV_i of the integration points. Available forms include for example Gaussian integration and trapezoidal integration. (Monte-Carlo integration may also be considered as an interpretation of "integration" in collocation MMs).

Gaussian integration rules are most frequently used for the integration in MMs. They integrate polynomials of order $2n_q - 1$ exactly where n_q is the number of quadrature points per dimension. The special weighting of this rule makes only sense if the meshfree integrands (being a product of a test and trial function) are sufficiently polynomial-like in the integration domains. It should be noted that even if the meshfree shape functions are sufficiently polynomial-like in their *supports*, this may not be the case for the integrand in the *integration domain*.

Assume, for example, the evaluation of the matrix element A_{IJ}, being an integral over some product of the test function N_I with the shape function N_J. Both are assumed polynomial-like in their supports $\widetilde{\Omega}_I$ and $\widetilde{\Omega}_J$, and it may be concluded that this is also the case in the intersection of these supports $\widetilde{\Omega}_I \bigcap \widetilde{\Omega}_J$, see also Fig. 12. However, choosing a different (larger) integration domain than this intersection necessarily leads to parts which

are 0, and the integrand in the integration domain may no longer be sufficiently polynomial-like (it is then only C^0-continuous). Then, the use of Gaussian integration rules may be unsuited and the trapezoidal rule may be preferred.

Direct Nodal Integration. Evaluating the integrals only at the nodal positions $\boldsymbol{x}_i$ instead of introducing integration points is called direct nodal integration. The resulting meshfree method is closely related to collocation MMs [17, 19]. The integration is clearly substantially more efficient than using full integration. However, in addition to comparatively large integration errors, a stability problem arises for the direct nodal integration which is very similar to some numerical problems in collocation methods. Stabilization approaches have been proposed to overcome these problems [15, 37].

Even for stabilized nodal integration schemes, the accuracy—in reference to convergence rate and absolute accuracy—of nodally integrated Galerkin MMs is considerably lower than for full integration, see e.g. [19] for a comparison.

Integration with Background Mesh or Cell Structure. In these techniques, the domain is divided into integration subdomains which are very similar to a mesh. However, this mesh has neither to be conforming nor aligning with, for example, the boundaries. The resulting MMs are often called pseudo-meshfree as only the approximation is truly meshfree, whereas the integration requires some kind of mesh. In case of a background mesh, nodes and integration cell vertices coincide in general—as in conventional FEM meshes. In case of integration with a cell structure, nodes and integration cell vertices do in general not coincide at all [44]. The difference may be seen in Fig. 11.

The problem of background meshes and cells is that the integration error which arises from the misalignment of the supports and the integration domains is often higher than the one which arises from the non-polynomial character of the shape functions [44]. It is pointed out that in case of the FEM, supports and integration domains always coincide.

Special techniques, such as those proposed in [44], construct integration cells that align with the shape functions supports by means of a bounding box technique. This approach is very closely related to integration over intersections of supports as discussed below. The use of adaptive integration by means of adaptively refining the mesh (which does not have to be conforming) or cell structure is discussed in [173].

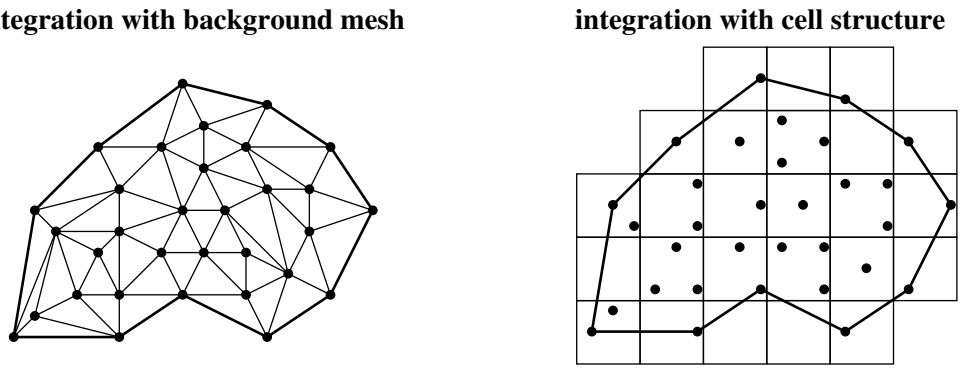

Figure 11: Integration with background mesh or cell structure.

Integration over Supports or Intersections of Supports. In this method, the domain of integration is directly the support of each node or even each intersection of the supports respectively, see Fig. 12. The resulting scheme is truly meshfree. The results in case of integrating over intersections of subdomains are much better than in the mesh or cell-based integration of the pseudo-meshfree methods for the same reason as in the above mentioned closely related alignment technique.

From an implementational point of view it should be mentioned that the resulting system matrix is integrated line by line and no element assembly is employed. For the integration over supports the integration points are distributed individually for each *line* of the final matrix, whereas the integration over intersection of supports distributes integration points for each *element* of the final matrix individually.

4.3.7 Discontinuities

The continuity of meshfree shape functions is often considerably higher than for FEM shape functions. In fact, they can be built with any desired order of continuity depending most importantly on the choice of the weighting function, see section 4.3.3. The resulting derivatives of meshfree interpolations are also smooth leading in general to very desirable properties, like smooth stresses etc. However, many practical problems involve physically justified discontinuities. For example, in crack simulation the displacement field is discontinuous, whereas in a structural analysis of two different connected materials the stresses are discontinuous. In the prior case the discontinuity is related to the interpolation itself, in the latter case only to the derivatives (discontinuous derivatives occur whenever the

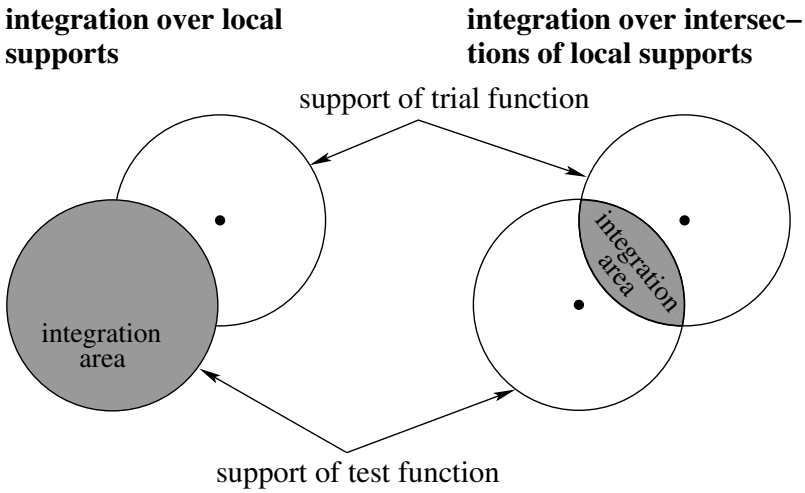

Figure 12: Integration over local supports or intersections of local supports.

coefficients of the PDE under consideration are discontinuous).

MMs need certain techniques to handle these discontinuities. Classical meshbased methods have problems to handle these problems, because there the discontinuity must align with element edges; although also for these methods ways have been found to overcome this problem (e.g. [23]). The treatment of discontinuities has similar features than the treatment of non-convex boundaries, see Fig. 13. It is cited from [79]:

> One has to be careful with performing MLS for a domain which is strongly non-convex. Here, one can think of a domain with a sharp concave corner. To achieve that MLS is well defined for such a domain and to have that the shape functions are continuous on the domain, it is possible that shape functions become non-zero on parts of the domain (think of the opposite side of the corner) where it is more likely that they are zero. Hence, nodal points can influence the approximant u^h on parts of the domain where it is not really convenient to have this influence.

Here, only methods are discussed which modify the supports according to the discontinuity, see [160] for an interesting comparison of these methods. Other methods such as those which incorporate discontinuous approximations as an enrichment of the basis functions [23, 25, 115] are not considered in the following.

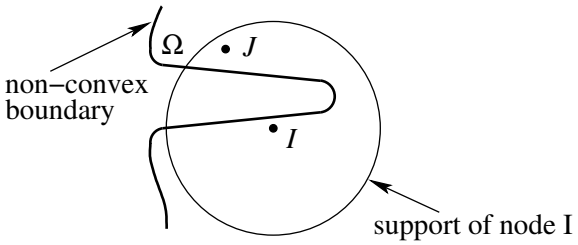

Figure 13: One has to be careful for non-convex boundaries. The support of node I should be modified, therefore, the same methods as for discontinuity treatment may be used.

Visibility Criterion. The visibility criterion [22] may be easily understood by considering the discontinuity opaque for "rays of light" coming from the nodes. That is, for the modification of a support of node I one considers light coming from the coordinates of node I and truncates the part of the support which is in the shadow of the discontinuity. This is depicted in Fig. 14a).

A major problem of this approach is that at the discontinuity tips an artificial discontinuity inside the domain is constructed and the resulting shape functions are consequently not even C^0-continuous. Convergence may still be reached [116], however, significant errors result and oscillations around the tip can occur especially for larger dilatation parameters [160]. The methods discussed in the following may be considered as improved versions with respect to the shortcomings of the visibility criterion and show differences only in the treatment around the discontinuity tips.

It shall further be mentioned that for all methods that modify the support—which in fact is somehow a reduction of its prior size—there may be problems in the regularity of the $k \times k$ system of equations, see section 4.3.4, because less supports overlap with the modified support. Therefore, it may be necessary to increase the support size leading to a larger band width of the resulting system of equations [25].

Diffraction Method. The diffraction method [20, 160] considers a diffraction of the rays around the tip of the discontinuity. For the evaluation of the weighting function at a certain evaluation point (usually an integration point) the input parameter of $\phi(\|\boldsymbol{x} - \boldsymbol{x}_I\|) = \phi(d_I)$ is changed in the following way: Define $s_0 = \|\boldsymbol{x} - \boldsymbol{x}_I\|$, s_1 being the distance from the node to the crack tip, $s_1 = \|\boldsymbol{x}_c - \boldsymbol{x}_I\|$, and s_2 the distance from the crack tip to the

evaluation point, $s_2 = \|\boldsymbol{x} - \boldsymbol{x}_c\|$, see Fig. 14b). Then, d_I is modified as [160]

$$d_I = \left(\frac{s_1 + s_2}{s_0} \right)^{\gamma} s_0; \tag{4.41}$$

in [20] only $\gamma = 1$, i.e. $d_I = s_1 + s_2 = \|\boldsymbol{x}_c - \boldsymbol{x}_I\| + \|\boldsymbol{x} - \boldsymbol{x}_I\|$, has been proposed. Reasonable choices for γ are 1 or 2 [160], however, optimal values for γ are not available and are problem specific. The derivatives of the resulting shape function are not continuous directly at the crack tip, however, this poses no difficulties as long as no integration point is placed there [160].

The modification of the support according to the diffraction method may be seen in Fig. 14b). A natural extension of the diffraction method for the case of multiple discontinuities per support may be found in [152].

Transparency Method. In [160] the transparency method is introduced. Here, the function is smoothed around a discontinuity by endowing the surface of the discontinuity with a varying degree of transparency. The tip of the discontinuity is considered completely transparent and becomes more and more opaque with increasing distance from the tip. For the modification of the input parameter of the weighting function d_I follows

$$d_I = s_0 + \rho_I \left(\frac{s_c}{\overline{s}_c} \right)^{\gamma}, \gamma \geq 2, \tag{4.42}$$

where $s_0 = \|\boldsymbol{x} - \boldsymbol{x}_I\|$, ρ_I is the dilatation parameter of node I, s_c is the intersection of the line $\overline{\boldsymbol{x}\boldsymbol{x}_I}$ with the discontinuity and $\overline{s}_c$ is the distance from the crack tip where the discontinuity is completely opaque, see Fig. 14c). For nodes directly adjacent to the discontinuity a special treatment is proposed [160]. The value γ of this approach is also a free value which has to be adjusted with empirical arguments. The resulting derivatives are continuous also at the crack tip.

4.4 Specific Meshfree Methods

In the following a brief description of some important specific MMs is given. In particular, collocation and Bubnov-Galerkin MMs are described, as well as MMs which employ an additional extrinsic basis for the approximation, see Fig. 4 on page 34. One may relate certain properties for each of these classes of MMs.

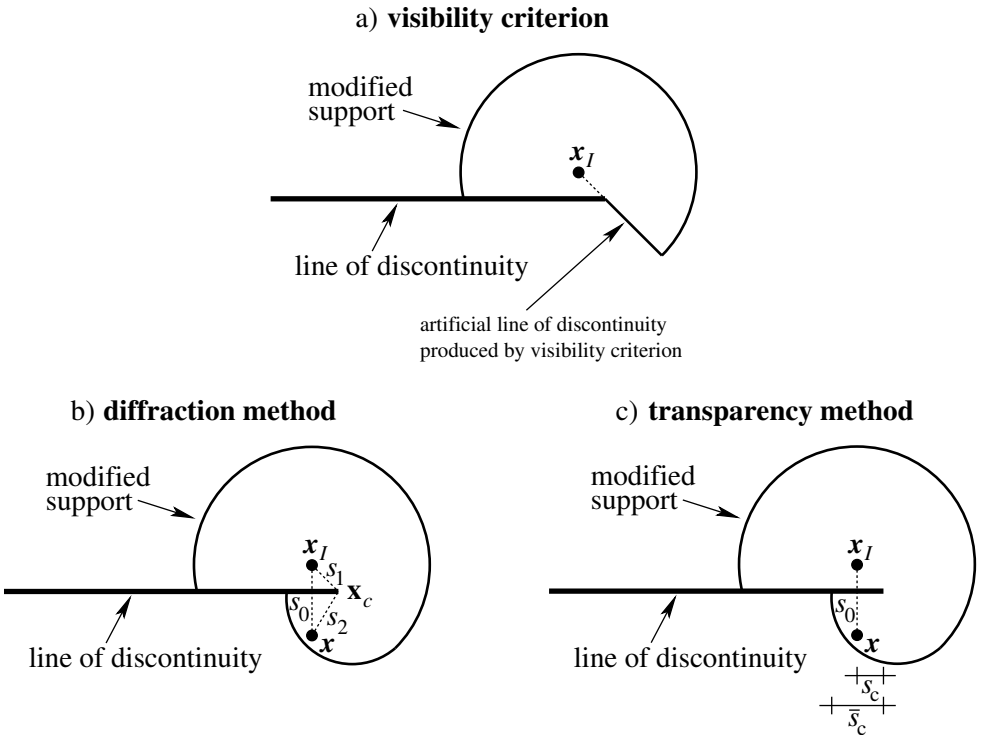

Figure 14: Visibility criterion, diffraction and transparency method for the treatment of discontinuities.

4.4.1 Collocation Meshfree Methods

The most well-known collocation MM is smoothed particle hydrodynamics (SPH). It was introduced in 1977 by Lucy in [142], and is often considered the first MM. The SPH is a Lagrangian collocation method, i.e. the collocation points (=particles) move with their associated velocity through the domain. This makes it a typical representative of "particle methods" [17, 68, 148]. It was first used for astronomical problems, and later extended to fluid (and structural) problems [68]. The basic idea of the SPH is a kernel approximation of the form (4.29), which is re-written here in its continuous and discrete form as

$$u^h(\boldsymbol{x}) = \int_{\Omega_{\boldsymbol{y}}} K(\boldsymbol{x},\boldsymbol{y})\, u(\boldsymbol{y})\, d\Omega, \tag{4.43}$$

$$= \sum_{i=1}^{r} K(\boldsymbol{x},\boldsymbol{x}_i)\, u(\boldsymbol{x}_i)\, \Delta V_i. \tag{4.44}$$

The kernel $K(\boldsymbol{x},\boldsymbol{y})$ is replaced by Gaussian-like functions such as those discussed in section 4.3.3. It becomes obvious that the name SPH stems from the smoothing character of the particles' point properties to the kernel function, thus leading to a continuous field. However, a simple discrete kernel of the form $K(\boldsymbol{x},\boldsymbol{x}_i) = \phi(\boldsymbol{x}-\boldsymbol{x}_i)$ is in general not able to fulfill consistency requirements of even 0-th order, see e.g. [21]. This leads to major drawbacks of the traditional SPH concerning stability and accuracy [17, 43, 177]. These effects are most serious near the boundary, see e.g. [32, 125] ("spurious boundary effects"). Fixing the lack of consistency by a kernel which is able to fulfill the reproducing conditions of any desired order leads to the reproducing kernel particle methods (RKPM)—in a collocation setting—, see section 4.3.1. Also a number of other fixed versions of the SPH exists, which are all able to fulfill consistency requirements of at least 0-th order. The corrected SPH (CSPH) [28, 120] and moving least-squares SPH (MLSPH) [43] are popular examples. See [19], [125] and [164] for an overview and further references for the various fixing approaches for the shortcomings of traditional SPH.

The finite point method (FPM), introduced by Oñate *et al.* in [159], is a consistent collocation method which is based on fixed (Eulerian) particles in contrast to the moving (Lagrangian) particles of the SPH. It employs shape functions generated by different least-squares procedures, including MLS.

A problem of collocation methods is that there is no systematic way to handle neither rigid nor moving boundaries [136]. According to [149], rigid walls have been simulated using

(a) forces with a length scale h (this mimics the physics behind the boundary condition), (b) perfect reflection, and (c) a layer of fixed particles. The fixed particles in the latter approach are often called ghost particles, see e.g. [164], where boundary conditions in SPH have been intensively discussed. Natural boundary conditions are also a major problem in SPH and collocation methods in general [17].

4.4.2 Bubnov-Galerkin Meshfree Methods

Using meshfree shape functions in a Bubnov-Galerkin setting was first presented by Nayroles *et al.* in [154]. They call their method diffuse element method (DEM) and construct the shape functions as a generalization of the FEM procedure. A possible way to obtain finite element shape functions is to employ a weighted least-squares method, where the weighting functions are constant over selected subdomains defined by means of a mesh. In the DEM, these subdomains (elements) are replaced by *diffuse* overlapping elements leading to exactly the same shape functions as those of the MLS method (although this was not realized by Nayroles *et al.* [114]). Although the shape functions of the DEM are identical to the MLS shape functions, Nayroles *et al.* made a number of simplifications:

- The derivatives are not computed correctly according to (4.26) and (4.27), but under the assumption that $\boldsymbol{a}(\boldsymbol{x})$ is constant. It is shown in [114] that the resulting derivatives are no longer integrable and consequently do not fulfill a requirement on the test functions in a weighted-residual method.
- Only very low quadrature rules for integration are used [141]. Then, it is unlikely that a sufficiently accurate integration of the Bubnov-Galerkin weighted residual is evaluated.
- Essential boundary conditions are not enforced correctly [141].

The element-free Galerkin (EFG) method, proposed in [22] by Belytschko *et al.*, corrects these simplifications. That is, the derivatives of the MLS functions are computed correctly, sufficiently large number of integration points are used, and essential boundary conditions are enforced correctly (in [22] by Lagrangian multipliers). Throughout this paper, the term EFG is used for a MM that employs MLS shape functions in a Bubnov-Galerkin or Petrov-Galerkin setting as those resulting from stabilized formulations, see section 5. The

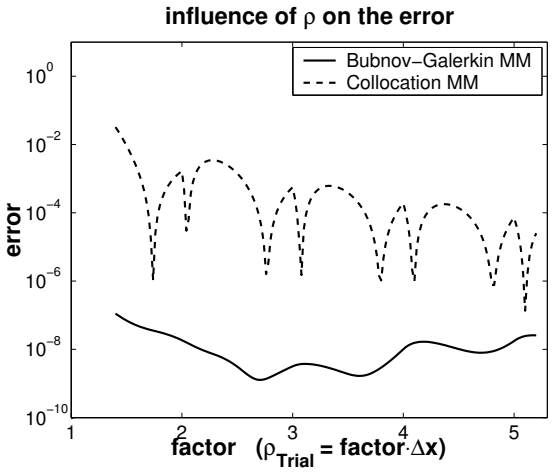

Figure 15: L_2-error for Bubnov-Galerkin and collocation MMs in dependence of varying dilatation parameters.

integration method or the enforcement of essential boundary conditions are not considered to be identifying parts of the method.

The EFG is one of the most popular MMs in practice. In comparison with collocation MMs, described in the previous subsection, Galerkin MMs such as the EFG are found to be more accurate but also more time-consuming due to the large number of integration points, which are necessary for the integration of the weak form. In [118], the authors claim that the EFG (=MLS) shape functions are 50-times more expensive to compute than FEM shape functions, and our own experiences confirm this statement.

The higher accuracy of Galerkin methods compared to collocation methods is demonstrated with an example, see Fig. 15. The one-dimensional advection-diffusion equation $c \cdot u_{,x} - K \cdot u_{,xx} = 0$ on $\Omega = (0,1)$ with $u(0) = 0$ and $u(1) = 1$ is approximated by a Bubnov-Galerkin and collocation MM. 21 regularly distributed nodes are employed and the advection-diffusion ratio is chosen small enough such that stabilization of this problem is not necessary. The dilatation parameter varies between $1.3\Delta x \leq \rho \leq 5.3\Delta x$. It may be seen that Bubnov-Galerkin MMs are more accurate than collocation MMs and tend to be less sensitive in the dilatation parameter. For an explanation for the rise and falls of the plot in dependence of the dilatation parameter, see [134].

4.4.3 Meshfree Methods with Extended Approximation Properties

The partition of unity FEM (PUFEM) [145], generalized FEM (GFEM) [173, 174], extended FEM (XFEM) [23] and the partition of unity methods (PUM) [10] may be considered to be essentially identical methods, following e.g. [7, 8]. Even those methods which have the term "finite element" in their name do not necessarily rely on a meshbased PU (although this might have been the case in the first publications of the method). All these methods employ an *extrinsic* basis $\boldsymbol{q}(\boldsymbol{x})$ in the approximation

$$u^h(\boldsymbol{x}) = \sum_{i=1}^{r} N_i(\boldsymbol{x})\, \boldsymbol{q}^{\mathrm{T}}(\boldsymbol{x})\, \boldsymbol{v}_i; \tag{4.45}$$

the *intrinsic* basis $\boldsymbol{p}(\boldsymbol{x})$ is used for the construction of a PU $\{\boldsymbol{N}\}$, which may either be obtained in a meshfree or meshbased way. Instead of having only $\boldsymbol{u}$ as unknowns one has l times more unknowns $\boldsymbol{v}_i = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_l)$. $\boldsymbol{q}(\boldsymbol{x})$ may consist of monomials, Taylor polynomials, Lagrange polynomials or any other suitable functions. For example, Babuška and Melenk use in [9] for the Helmholtz equation in one dimension the extrinsic basis $\boldsymbol{q}^{\mathrm{T}}(x) = \left[1, x, \ldots, x^{l-2}, \sinh nx, \cosh nx\right]$.

Some main features of this class of methods are:

- A lower order consistent PU can be enriched to a higher order approximation [10]. For example a zero-order PU may be extended to higher-order consistency by choice of a suitable extrinsic basis. In this case, the number of unknowns per node is l-times higher, leading to large system of equations. In contrast, constructing a higher-order PU with the MLS procedure from the beginning leads always to only one unknown per node and the increased work lies in the inversion of larger moment matrices.

- *A priori* knowledge about the solution can be included into the approximation, and thus the trial and test spaces can be designed with respect to the problem under consideration [7, 10, 145, 173, 174]. Standard FEMs and MMs rely on the *local* approximation properties of polynomials, being used in the intrinsic basis. However, if—from analytical considerations—the solution is known to have locally a non-polynomial character (e.g. it is oscillatory, singular etc.), the approximation should better be done by ("handbook") functions that are more suited than polynomials, e.g. harmonic functions, singular functions etc., in order to gain optimal convergence properties.

- One can easily construct ansatz spaces of any desired regularity, while in the FEM it is a severe constraint to be conforming. The approximation properties of the FEM are based on the local approximability and the fact that polynomial spaces are big enough to absorb extra constraints of being conforming without loosing the approximation properties. Instead, in the PUM, the smoothness of the PU enforces easily the conformity of the global space and allows one to concentrate on finding good local approximations for a given problem [145].

- The essential boundary conditions can be implemented by choosing the local approximation spaces such that the functions satisfy them [7, 8]. In contrast, standard MMs based on the MLS or RKPM procedure without an additional extrinsic basis require special attention for the imposition of EBCs, see section 4.3.5.

The aspect of including *a priori* knowledge into the approximation space is often the most important reason for using PUMs in practice. For standard problems, polynomial approximation spaces are most often well-suited. Then, standard MMs which use PU-functions straightforward for the approximation are the first choice.

4.5 Choice of a Meshfree Method for Flow Problems

A number of different MMs and related problems have been discussed in the previous subsections. In this subsection, the particular MM chosen for the approximation of fluid and fluid-structure-interaction problems in the sequel of this work is described in detail. Most MMs for the simulation of flow problems are Lagrangian collocation methods such as the SPH, see section 4.4.1. These methods are comparatively fast, but the accuracy is often a problem. Boundary conditions are not easy to apply in general, especially at the outflow, where particles leave the domain.

The accuracy aspect is closely related to the use of a collocation setting, which is considerably less accurate for the same number of nodes than an equivalent Galerkin method, see e.g. Fig. 15. The problem with the boundary condition treatment is, in addition, related to the Lagrangian formulation. Particles move through the domain and it requires special techniques to apply the boundary conditions at certain positions.

Therefore, the focus in this work is on meshfree Eulerian and ALE Galerkin methods. Using a Galerkin method promises high accuracy, whereas the Eulerian and ALE formulation

offers the advantage that particles remain constant in most parts of the domain. Particles are placed at suitable positions, e.g. along the boundary, and remain there throughout the calculation. The enforcement of boundary conditions is comparably easy and accurate. The fact that particles stay at positions where this is desirable has also advantages in adaptive procedures.

However, Eulerian and ALE Galerkin MMs are not without shortcomings. Using Eulerian and ALE formulations for advection-dominated problems, such as those which frequently arise in the context of fluid dynamics, requires *stabilization*. This is well-known in the meshbased context, where various stabilization strategies have been developed. In section 5, some of these techniques are extended for the use in the meshfree context. Stabilized MMs are a crucial aspect for the success of Eulerian and ALE MMs.

Another problem of Eulerian and ALE Galerkin MMs is the increased amount of computational work—compared with collocation methods—associated with the integration of the weak form. Especially in flow simulation often large numbers of unknowns are involved, and the integration may be prohibitively expensive. Therefore, it seems promising to *couple* standard meshbased methods like the FEM with meshfree Galerkin methods. Then, it is possible to use MMs only in small regions of the domain, where a mesh is difficult to maintain, and meshbased standard methods in all other parts. The computational work is expected to scale with the meshbased part, rather than with the meshfree one. It should be noted that Eulerian and ALE Galerkin methods are standard in finite element simulations of flow problems, and it is therefore a natural and straightforward choice to couple them with their meshfree counterparts. The aspect of coupling is worked out in section 6.

The individual aspects of the chosen MM according to the previous subsections are described as follows:

Shape and test functions. (section 4.2.2–4.2.4) The standard MLS functions are chosen to construct a PU of first order, which is the minimum requirement for the Navier-Stokes equations. The approximation relies on these functions only and does not employ an additional extrinsic basis, which is, in practice, only chosen for very special problems. The test functions are chosen according to the stabilization method which is involved, leading to Petrov-Galerkin settings, see section 5. For all problems which do not require stabilization, a simple Bubnov-Galerkin approach is taken, that is, the test functions are identical to the shape functions.

This method is equivalent to the element-free Galerkin (EFG) method, which in its original version is used in Bubnov-Galerkin settings only, together with a certain treatment of boundary conditions and integration. In this work, Petrov-Galerkin methods are frequently used, and integration and boundary condition aspects are realized differently. Nevertheless, the most important idea of the EFG remains: the use of MLS shape functions in Galerkin settings.

Weighting function. (section 4.3.3) The forth-order spline is used as the standard weighting function for all numerical studies presented herein. Also many other possibilities have been realized successfully in our own numerical studies, however, no significant advantages over the forth-order spline could be noted for other weighting functions. And the computational effort for the evaluation of the different weighting functions varies considerably, with the forth-order spline being one of the least expensive alternatives. In more than one dimension, tensor products as well as spherical generalizations of the forth-order spline are employed.

Boundary conditions. (section 4.3.5) The boundary collocation method seems to be the simplest approach and is employed in a number of test cases. A possible loss of convergence for higher order approximations was not noted for the first order consistent shape functions chosen herein. Also the penalty method is employed in one test case (section 6.5.6). In the coupled meshfree/meshbased fluid solver, see section 6, boundary conditions are prescribed by help of coupling finite elements along the boundary.

Integration. (section 4.3.6) Both, background integration by means of a mesh and integration over supports are realized. Gaussian and trapezoidal rules with different numbers of integration points may be taken for the numerical evaluation of the integrals.

Discontinuities. (section 4.3.7) The visibility criterion is realized for a test case described in section 6.5.6. It is claimed in [160] that this criterion leads for small dilatation parameters ρ to almost identical results as obtained by the diffraction and transparency method.

5 Stabilization of Meshfree Methods

It has been discussed in section 2.1 that one may consider different formulations of the underlying partial differential equations. Most importantly Lagrangian, Eulerian, and arbitrary Lagrangian-Eulerian (ALE) formulations [98] have to be mentioned, which choose distinct reference systems for the description of the problem. The most important difference in the formulations is in the presence of an advection term in the Eulerian and ALE formulations, which is absent in the Lagrangian description. Advection terms are non-selfadjoint operators that often lead to problems in their numerical treatment, e.g. [31]. This is particularly the case for Bubnov-Galerkin methods, where the test functions are chosen equal to the shape functions. Spurious oscillations may pollute the overall solution and stabilization is required, see e.g. [31].

Numerical problems (locking, singular matrices etc.) may also occur with so-called mixed problems [55], the incompressible Navier-Stokes equations fall into this class. Applying the same shape functions to all variables of the problems in a Bubnov-Galerkin setting (equal-order interpolation), which is from a computational viewpoint the most convenient way, leads to severe problems as a result from violating certain conditions. Stabilization is a possibility to overcome these problems [55].

The need for stabilization is well studied in the meshbased context, e.g. [31, 46, 97]. A number of stabilization techniques have been developed to overcome numerical problems. This stems from the fact that for meshbased methods the Eulerian or ALE formulation is standard, e.g. [46, 82], because it seems impossible to maintain a conforming mesh in most flow problems with the Lagrangian formulation. Then, stabilization is a crucial ingredient to obtain suitable solutions.

Eulerian and ALE meshfree methods are not only of interest in their own right, but are a natural choice for the desired coupling of meshbased and meshfree methods. So far, MMs for fluid simulation have usually been used in Lagrangian formulations, i.e. as particle methods [112, 148, 149, 182]. To successfully use Eulerian or ALE meshfree methods in Galerkin settings, the problem of stabilization has to be solved. This is the main aspect of this section.

The focus is in particular on two standard stabilization schemes known from the meshbased context, which are frequently applied to the incompressible Navier-Stokes equations: One is the streamline-upwind/Petrov-Galerkin method [31] together with the pressure-

stabilizing/Petrov-Galerkin method [179] (SUPG/PSPG), the other is the Galerkin/least-squares method [97] (GLS). Analogously to most stabilization techniques for Galerkin methods, both methods add products of perturbation terms with residual terms of the governing problem to the weak form, weighted with a stabilization parameter τ [60]. They smooth oscillations in convection-dominated regimes, and overcome problems associated with equal-order interpolations in mixed formulations. It is found that SUPG/PSPG as well as GLS stabilization can be applied straightforward to meshbased and meshfree problems. However, the aspect of the stabilization parameter τ has to be reconsidered: Standard formulas for τ are often deduced for linear finite element shape functions. Applications to higher order elements as well as to meshfree shape functions requires special attention [53, 54, 60, 63].

Section 5.1 describes the need for stabilization and gives a review of the development of stabilized methods. SUPG, PSPG and GLS stabilization are defined in section 5.2. The aspect of suitable stabilization parameters τ is discussed in detail in the following section. Finally, the numerical results in section 5.4 show the successful extension of stabilization to meshfree Galerkin methods.

5.1 The Need for Stabilization

5.1.1 Convection-dominated Problems

The phenomenon of *convection*, typically identified by first order terms in the differential equations of a model, divides the usability of numerical methods. Methods being successfully applied in structure problems, where no convection is present, may totally fail when they are applied to convection-dominated problems, as they occur frequently e.g. in fluid mechanics. This is particularly the case with Bubnov-Galerkin methods [31]. Fig. 16 shows an oscillatory example for a meshbased and a meshfree Bubnov-Galerkin approximation of a convection-dominated problem (one-dimensional advection-diffusion equation).

In structural analysis, where often the minimization of energy principles is the underlying idea, the application of Bubnov-Galerkin methods leads to symmetric matrices and "optimal" results. "Optimal" refers to the fact that the solution often possesses the best approximation property, meaning that the difference between the approximate and the exact solution is minimized with respect to a certain norm [31].

The situation, however, is totally different in the presence of convective terms. Then,

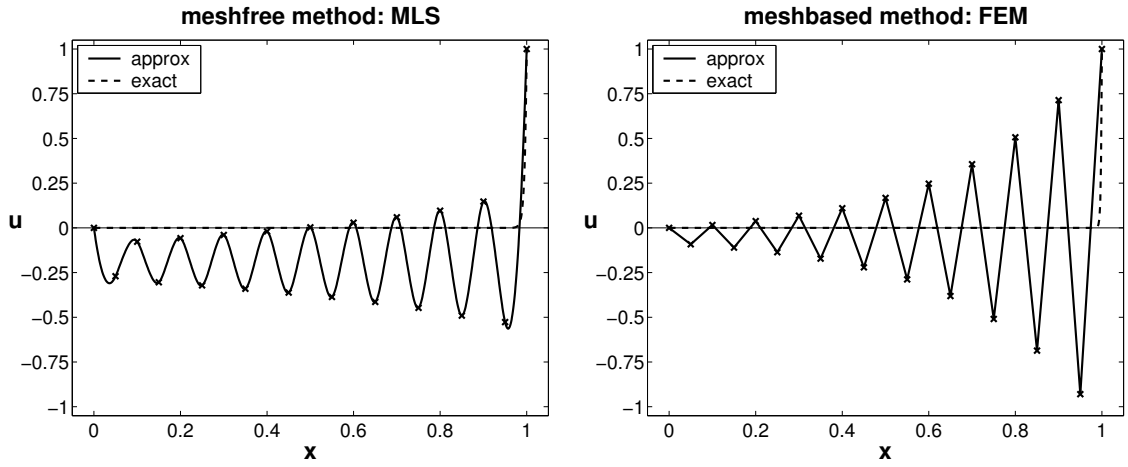

Figure 16: Typical oscillations in the approximation of an advection-dominated problem in one dimension.

the matrix associated with the advective term is non-symmetric (non-self adjointness of the convective operator) and the best approximation property is lost [31]. As a result Bubnov-Galerkin methods applied to these problems are far from optimal and show spurious oscillations in the solutions. The situation gets worse for growing convection-diffusion ratios, defined by well-known identification numbers such as the Peclet and Reynolds number. The higher these numbers are, the more dominant is the convection term and the stronger is the pollution with oscillations. This does not only lead to qualitatively bad results but even violates basic physical principles like entropy production [82] or the positive boundedness of concentrations etc.

The same situation can be found in the finite difference context. There, the problem with oscillations occurs when using central differences for the advective operator. It can be easily shown that Bubnov-Galerkin treatment of the weak form and central differences applied to the strong form are closely related. For example, the corresponding matrix line of node I of a one-dimensional advective operator $cu_{,x}$ becomes for linear FEM and FDM in case of a regular node distribution:

$$\text{FEM}: \quad \tfrac{c}{2}\left[\ldots -1, 0, +1 \ldots\right] \qquad \text{FDM}: \quad \tfrac{c}{2\Delta x}\left[\ldots -1, 0, +1 \ldots\right] \ . \tag{5.1}$$

In the FDM context, it is well-known that upwind differencing on the convective term does not show oscillatory solutions, but introduces over-diffusive results [31]. A simple Taylor series analysis proves that upwinding is only first order accurate, in contrast to the second

order accurate—but oscillatory—central differences. This analysis also elucidates that upwinding can also be interpreted as central differences plus artificial diffusion. Thus, the "right" combination of central and upwind differences may introduce the optimal amount of artificial diffusion which leads to accurate and oscillation-free solutions [31].

In the seventies, a large number of FEMs were proposed with different ideas to include the upwind effect in finite elements, see [42, 80, 91, 110]. Often, the one-dimensional advection-diffusion equation served as a relevant model equation and generalization to other problems in multi-dimension was straightforward—but unsuccessful. The proposed methods obtained nodally exact solutions for the one-dimensional model equation, such that the resulting difference stencil of the FEM matches exactly the known nodally exact stencil from the FDM. This was realized in the "anisotropic balancing dissipation" approach by adding artificial diffusion in streamline direction and using a standard Bubnov-Galerkin method to discretize the modified problem [110]. Thereby, the consistency of the method is given up, i.e. the exact solution does not longer fulfill the modified weak form. Other approaches used a Petrov-Galerkin FEM, where the test functions are modified such that they weigh the upwind node more than the downstream node, see e.g. [42, 80]. In [91], the advection term is integrated with only one integration point, which is placed inside the element in dependence of the convection-diffusion ratio, whereas all other terms are integrated in the standard way. All these approaches obtain the optimal difference stencils in the resulting system of equations leading to the nodally exact solution for the one-dimensional model problem. However, successful generalization to arbitrary, time-dependent problems and multi-dimensions failed—i.e. the results were either oscillatory or over-diffusive—, and a successful method was still outstanding.

The Streamline-Upwind/Petrov-Galerkin (SUPG) method, introduced in [31] (and [94]) by Brooks and Hughes may be considered as the first successful stabilization technique to prevent oscillations in convection-dominated problems in the FEM. The main steps are [31, 60]: Introduce artificial diffusion in streamline direction only, interpret this as a modification of the test function of the advection terms and finally, enforce consistency, such that this modified test function is applied to *all* terms of the weak form. Then, the term "artificial diffusion" is not fully applicable any longer, because the stabilized weak form can not, in general, be manipulated such that only a diffusion term is extracted. The resulting SUPG stabilized weak formulation is still consistent, i.e. the exact solution of the problem still satisfies the stabilized equations. In the following, SUPG was successfully extended to coupled multidimensional advection-diffusion systems, where each equation

has to be stabilized. The Euler and Navier-Stokes equations also fall into this class, the first being the governing equations of inviscid flow, the latter of viscous flow [1, 102, 99]. Incompressible flows can be handled very successfully without stabilizing each equation individually [31]. The major part of the theoretical analysis of the SUPG was done by Johnson *et al.*, see [107, 153] and references therein. There, SUPG is often labeled with the term "streamline diffusion method".

Motivated from mathematical analysis, another type of stabilization scheme has been established, the Galerkin/Least-Squares (GLS) method [97]. It is similar to the SUPG in certain aspects, and for purely hyperbolic equations and/or linear interpolation functions, the two become identical [97]. In the GLS method, least-squares forms of the residuals are added to the Galerkin method, enhancing the stability of the Bubnov-Galerkin method without giving up consistency or degrading accuracy [97]. There is no motivation from artificial diffusion as was the starting point for SUPG. The GLS method was introduced under this name as a method on its own in [97] by Hughes, Franca and Hulbert. They apply the GLS method for stationary and instationary advective-diffusive systems. In [168], Shakib uses the GLS for the solution of the compressible Euler and Navier-Stokes equations.

Today, the SUPG and GLS stabilizations are most frequently used to stabilize FEM formulations, see e.g. [46]. Both stabilization methods add products of perturbations and residuals to the weak form, weighted with a so-called stabilization parameter τ. The suitable choice of τ, leading to reliable oscillation-free approximations, is a crucial aspect in stabilized methods [168].

Recently, stabilization has also been applied to meshfree methods [4, 75, 89, 119, 124, 133, 158, 159]. The same principles as for the FEM stabilizations have been used here. Upwind ideas for meshfree collocation methods—analogously to the meshbased FDM—have been examined e.g. by Kuhnert in [119], a different way is shown by Oñate *et al.* in [158, 159]. For meshfree Galerkin methods, upwind ideas have been investigated e.g. by Atluri *et al.* in [4]. There, the supports of the test functions are shifted in upstream direction depending on the local convection-diffusion ratio.

SUPG and GLS stabilized meshfree Galerkin methods have been successfully applied, e.g. by Huerta *et al.* in [89], by Liu *et al.* in [133] and by Li *et al.* in [124] for the solution of linear advection-diffusion problems. In [75], Günther applies SUPG stabilization to the compressible Navier-Stokes equations. It is not surprising that SUPG and GLS stabilizations work successfully for meshbased *and* meshfree methods as well, because close

similarities in the theoretical analysis can be shown, see e.g. [7, 8]. Therefore, one may expect that the theoretical foundation of SUPG and GLS accomplished for the meshbased FEM applies analogously to meshfree methods.

However, this is not generally true for the stabilization parameter τ. It is shown in section 5.3 that the resulting formulas for the parameter τ are highly dependent on the shape functions of the approximation. The standard formulas for τ used in the meshbased FEM context are derived for linear finite element shape functions. Application of these formulas to MMs does in general not lead to suitable results; this is also noted e.g. in [133] and [75]. In section 5.3 it is shown under which circumstances standard meshbased formulas for τ applied to MMs are suitable.

5.1.2 The Babuška-Brezzi Condition

Variational formulations associated with constraints lead to severe problems if standard numerical methods are used in a straightforward manner. One way to treat these problems is to use admissible functions satisfying the constraint *ab initio* [55]. The solution is then a member of a smaller space of functions than the space required from continuity conditions alone, but suitable interpolations are not easy to find. Instead, the problem can be reformulated by introducing a second variable, the Lagrange multiplier [55]. The resulting variational formulation falls into an abstract class of mixed formulations. Lagrange multipliers and mixed formulations are thus intimately related.

One of the most well-known examples of a mixed problem is Stokes flow, the non-advective counterpart of the incompressible Navier-Stokes equations, in which the velocity-strain energy is minimized subject to the incompressibility constraint. Fig. 17 shows an example for Stokes flow with large oscillations in the pressure field as a consequence of violating the Babuška-Brezzi condition.

The approximation of mixed formulations requires careful choice of the combination of interpolation functions. In particular, equal-order interpolations, where the same ansatz is made for the primary and secondary (Lagrange multiplier) variables are *not* adequate in a Bubnov-Galerkin setting, although from an implementational viewpoint they are most desirable [179]. Also, many other practically convenient interpolations fail to give satisfactory results [55], especially in three dimensions.

The governing stability conditions for mixed problems are *K-ellipticity* and the *Babuška-Brezzi condition* [6, 30]. Violating them leads to pathologies such as spurious oscillations

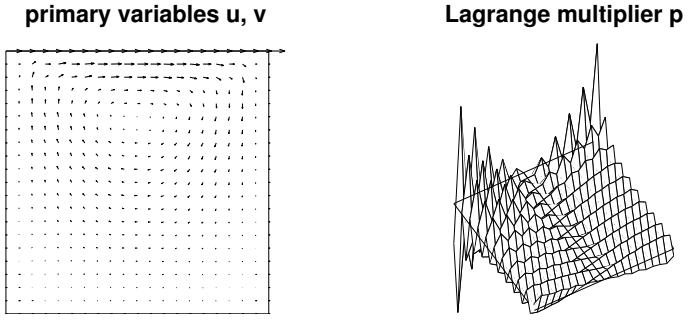

Figure 17: The solution for Stokes flow with P1/P1 FEM and a wrong stabilization parameter $\tau \approx 0$. Although the primary variables (velocity field) are reasonably approximated, the Lagrange multiplier (pressure field) shows large oscillations.

and locking [55], or the resulting system of equations may be singular not giving a solution at all. It depends on the concrete problem, which of the two criteria is more difficult to obtain [55]. Lack of stability may come from the Lagrange multiplier or from the primary variable. For problems, in which K-ellipticity is difficult to satisfy—e.g. for linear isotropic incompressible elasticity emanating from the Hellinger-Reissner principle—, the problem comes from the primal variable and it is often easy to find interpolations *satisfying* the Babuška-Brezzi condition.

In contrast, for problems that fulfill the ellipticity requirement immediately—like Stokes flow—, stability problems arise from the Lagrange multiplier and it is difficult to fulfill the Babuška-Brezzi condition. Only very few combinations of interpolations are adequate. In this case, it is desirable to find ways to *circumvent* the condition [55]. Motivated from theory this can be done by modifying the bilinear form such that it is coercive [190] on the primal variable as well as the Lagrange multiplier. Then, there is no need to fulfill the Babuška-Brezzi condition for this method. This can be interpreted as some kind of stabilization which is realized by adding appropriate perturbation terms, without upsetting consistency. It is realized—with the same fundamental idea as in other stabilizations—by a multiplication of perturbations with residual forms of the governing problem.

Stabilizations of the Stokes equations in the FEM context have been presented in [96], and in [179] for the incompressible Navier-Stokes equations. Both methods are very similar in that they only perturb the test function of the Lagrange multiplier, i.e. the pressure, leading to unsymmetric systems of equations for Stokes flow. This kind of stabilization is called pressure-stabilizing/Petrov-Galerkin (PSPG) throughout this paper, as proposed in

[179]. In [95], Stokes flow has been stabilized with GLS stabilization, leading to perturbations of all test functions but maintaining symmetry. GLS was already mentioned in the previous subsection for the stabilization of convection-dominated problems, and can also be used here to circumvent the Babuška-Brezzi condition. This is not the case for SUPG stabilization which is only successful in suppressing oscillations from convection-dominated problems. Thus, in the following GLS stabilization is considered on the one hand, and SUPG/PSPG stabilization on the other.

In a meshfree context, the aspects of mixed problems and problems with constraints have been investigated e.g. by Huerta *et al.* in [90], where a pseudo-divergence-free interpolation space is defined, enabling to fulfill the Babuška-Brezzi condition. There it is also pointed out that incompressibility in meshfree methods is still an open question. Related to finite elasticity and locking, the contributions in [38] and [87] are mentioned, the pressure projection method in the former is also known in the FEM context for the incompressible Navier-Stokes and Stokes equations, e.g. [139]. Stabilization of these problems in a meshfree context, equivalent to PSPG stabilization, may be found in [124] for Stokes flow.

5.1.3 Steep Solution Gradients

In section 5.1.1 it is discussed that convection-dominated problems require stabilization such that a pollution of the overall solution with oscillations is prevented. However, these stabilizations do not preclude "over- and undershooting" about sharp internal and boundary layers [101]. These somehow localized (in that they do not influence the whole domain) oscillations can be suppressed by getting control over the solution gradient. The aim is to obtain a *monotone* solution without any oscillations. These methods have also been called maximum-principle satisfying methods in the literature [147].

There is, however, a very severe restriction concerning the monotonicity of a numerical scheme, which is summarized in the theorem of Godunov. There, it is proven that no linear higher-order method can obtain monotone solutions [82]. Thus, there are only two ways to achieve monotonicity: Using first order accurate schemes such as upwind finite differences or using non-linear schemes. The first way is in fact no real alternative, as higher-order accuracy is essential in the reliable simulation of many problems, consequently non-linear schemes have to be developed.

In the resulting schemes, there is always some kind of analysis and control of an interim solution. In the finite difference and finite volume context this can for example be real-

ized with the so-called slope-limiter methods, a subclass of the monotone total variation diminishing (TVD) schemes [82]. The minmod-slope-limiter, Roe's superbee limiter, van-Leer-limiter are well-known examples of slope-limiters.

One of the first monotone methods in the finite element context for convection-diffusion problems is the one proposed in [165], where the non-linearity is introduced by detection of element downstream nodes and a specific element matrix for the advection term depending on that node. In [147], Mizukami and Hughes introduce a consistent monotone *Petrov-Galerkin* FEM, valid for linear triangular elements with acute angles only. In Petrov-Galerkin FEMs the non-linearity lies in the dependence of the perturbed test function upon the solution gradient. The resulting discretized equations are non-linear even for a linear problem.

In [101] over- and undershoots are stabilized with a discontinuity-capturing term, which is generalizable to complex multidimensional systems. This Petrov-Galerkin method contains test functions modified with the added discontinuity-capturing term, acting in the direction of the solution gradient. In contrast, the stabilizations of 5.1.1 act in the direction of the streamline. Having control in direction of the streamline and of the solution gradient enables higher-order monotone schemes with enhanced robustness with the price of non-linearity. In [181] the discontinuity operator is generalized to non-linear convection-diffusion-reaction equations and in [100] and later [168] to multidimensional advective-diffusive systems such as the Navier-Stokes equations.

It should be mentioned that a compromise has to be made, whether the steepness of a solution, or the monotonicity is of higher importance. It is an immanent feature of the shape functions of a numerical method—e.g. their supports and functional form—that only a certain gradient can be represented without over- and undershoots. This can be seen from Fig. 18. The only way to obtain a monotone solution is to smear out the steep gradients in the domain such that the method can represent it without over- and undershoots. A more accurate solution in terms of approximation errors is often obtained in the presence of over- and undershoots, i.e. without or tuned influence of a discontinuity-capturing term.

5.2 Stabilization Methods

In this section some of the most important stabilization methods are described. The aim is to outline their different structures and for which kind of problems—referring to section

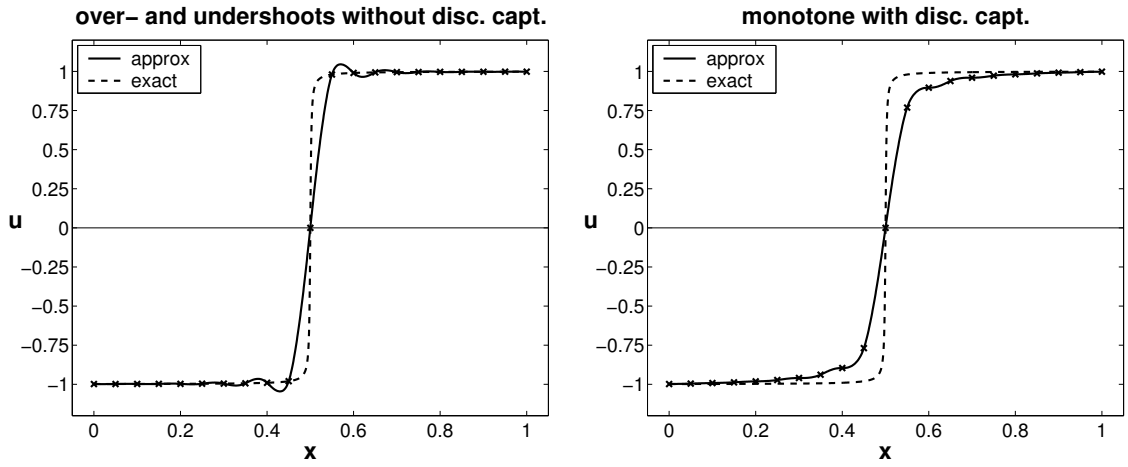

Figure 18: Approximation with over- and undershoots and monotone approximation.

5.1—they are suited. All stabilization schemes described in the following are Petrov-Galerkin approaches. They all add perturbations to the original Bubnov-Galerkin weak form. These perturbations are formulated in terms of modifications of the Bubnov-Galerkin test functions. They are multiplied with the residuals of the differential equations and thereby ensure consistency. Additionally, a stabilization parameter τ weighs the influence of the added stabilization terms.

5.2.1 Streamline-Upwind/Petrov-Galerkin (SUPG)

In section 5.1.1 it was claimed that, in the FDM, introducing artificial diffusion in a smart way smoothes out the oscillations in convection-dominated problems. This motivation is the starting point of the Streamline-Upwind/Petrov-Galerkin (SUPG) method [94, 31].

Motivation. To illustrate the motivation of SUPG stabilization the linear multidimensional advection-diffusion equation is considered as

$$\boldsymbol{c} \cdot \nabla u(\boldsymbol{x}) - \nabla \cdot (\mathbf{K} \nabla u(\boldsymbol{x})) = 0, \quad \forall \boldsymbol{x} \in \Omega \tag{5.2}$$

where $\boldsymbol{c}$ defines the advection and $\mathbf{K}$ is the diffusion tensor. Approximating the differential equation by a Bubnov-Galerkin method results in oscillations for large Peclet numbers.

Therefore, artificial diffusion is added in streamline-direction

$$\widetilde{\mathbf{K}}_{\xi\eta} = \begin{pmatrix} \widetilde{k} & 0 \\ 0 & 0 \end{pmatrix}, \tag{5.3}$$

with reference to the (ξ, η)-coordinate system, which is adjusted in streamline-direction. This diffusion tensor is projected into the standard coordinate system, where (5.2) is formulated by the transformation matrix $\mathbf{T}$ as

$$\widetilde{\mathbf{K}} = \mathbf{T}^{\mathrm{T}} \widetilde{\mathbf{K}}_{\xi\eta} \mathbf{T} = \frac{\widetilde{k}}{\boldsymbol{c} \cdot \boldsymbol{c}}\, \boldsymbol{c} \otimes \boldsymbol{c}. \tag{5.4}$$

The weak form of the problem becomes

$$\int_\Omega w \left(\boldsymbol{c} \cdot \nabla u - \nabla \cdot \left(\left(\mathbf{K} + \widetilde{\mathbf{K}} \right) \nabla u \right) \right) d\Omega = \mathbf{0}, \tag{5.5}$$

$$\int_\Omega w \boldsymbol{c} \cdot \nabla u + \nabla w : \left(\widetilde{\mathbf{K}} \nabla u \right) - w \nabla \cdot (\mathbf{K} \nabla u)\, d\Omega = \mathbf{0}, \tag{5.6}$$

$$\int_\Omega \left(w + \frac{\widetilde{k}}{\boldsymbol{c} \cdot \boldsymbol{c}} \boldsymbol{c} \cdot \nabla w \right) (\boldsymbol{c} \cdot \nabla u) - w \nabla \cdot (\mathbf{K} \nabla u)\, d\Omega = \mathbf{0}. \tag{5.7}$$

In (5.6), the divergence theorem is applied to the artificial diffusion term (neglecting boundary terms). Then, by using (5.4), the contribution of the artificial diffusion is interpreted as a modification of the test function of the advection term. Finally, in order to maintain consistency, this modified test function is applied to all terms of the differential equation

$$\int_\Omega (w + \tau \boldsymbol{c} \cdot \nabla w) \left(\boldsymbol{c} \cdot \nabla u - \nabla \cdot (\mathbf{K} \nabla u) \right) d\Omega = \mathbf{0}, \tag{5.8}$$

where $\tau = \widetilde{k} / (\boldsymbol{c} \cdot \boldsymbol{c})$ is called stabilization parameter.

Generalization. More general, for a discretized PDE of the form (3.3)

$$\int_\Omega \boldsymbol{w}^\star \left(\mathcal{L} u^h - f^h \right) d\Omega = \mathbf{0}, \quad u^h = \boldsymbol{N}^{\mathrm{T}} \boldsymbol{u}, \tag{5.9}$$

the SUPG-stabilized weak form follows from a streamline-upwind perturbation [31] of the test function as

$$\boldsymbol{w}^\star = \boldsymbol{w} + \tau \mathcal{L}_{\text{adv}} \boldsymbol{w}, \quad \boldsymbol{w} = \boldsymbol{N}. \tag{5.10}$$

$\mathcal{L}_{\text{adv}}$ is the advective part of the whole operator $\mathcal{L}$, for example, in case of the advection-diffusion equation, $\mathcal{L}_{\text{adv}} = \boldsymbol{c} \cdot \nabla$.

An important aspect is the consideration of the required continuity of the approximation spaces. In the FEM, piecewise polynomials are particularly useful shape functions, often having C^0-continuity in the domain Ω (and C^∞ inside an element). Then, the first derivatives include jumps at the element boundaries and second derivatives are Dirac-δ functions at the element boundaries. Integration in Ω over the product of two functions, where e.g. a jump and a Dirac-δ function coincide is not allowed (for example, this occurs in terms such as $\int_\Omega w_{,x} N_{,xx} d\Omega$). This problem is well-known in the context of the least-squares FEM [105], and may be handled there by using C^1-continuous shape functions, which may pose serious problems, see e.g. [127]. However, in the context of stabilization, where very similar terms as in the least-squares FEM occur, this problem is circumvented by defining the stabilization contributions *only inside element interiors*, where the shape functions are C^∞:

$$\int_\Omega \boldsymbol{w} \left(\mathcal{L} u^h - f^h \right) \, d\Omega + \sum_{\text{e}=1}^{n_\text{e}} \int_{\Omega_\text{e}} \tau \mathcal{L}_{\text{adv}} \boldsymbol{w} \left(\mathcal{L} u^h - f^h \right) d\Omega = \boldsymbol{0}. \tag{5.11}$$

Thereby, the stabilization does not upset higher continuity requirements as needed for the Bubnov-Galerkin weak form of the same problem. However, the under-representation of some terms (mainly diffusion, i.e. second order terms [31]) may result in a convergence order degradation in some cases [89]. Least-squares recovering techniques of second derivatives of the shape functions offer help, but the increase in computational cost is not negligible [89].

It should be noted that *meshfree* shape functions used in practice are always at least C^1-continuous—they can be constructed to have arbitrary continuity, see section 4.3.3—and that therefore no summation over subdomains of Ω has to be considered. Therefore, this aspect is not overemphasized throughout this work and it is written

$$\int_\Omega \left(\boldsymbol{w} + \tau \mathcal{L}_{\text{adv}} \boldsymbol{w} \right) \left(\mathcal{L} u^h - f^h \right) d\Omega = \boldsymbol{0} \tag{5.12}$$

for simplicity whenever the continuity consideration is of less importance.

SUPG weak form of the NS eqts. The Bubnov-Galerkin weak form of the incompressible Navier-Stokes equations is given in (3.17). The following terms are added for SUPG stabilization

$$\ldots + \sum_{\mathrm{e}=1}^{n_{\mathrm{el}}} \int_{\Omega_{\mathrm{F,e}}} \tau \left(\bar{\boldsymbol{u}}^h \cdot \nabla_{\boldsymbol{x}} \boldsymbol{w}^h \right) \cdot \left[\varrho_{\mathrm{F}} \left(\boldsymbol{u}^h_{,t} + \bar{\boldsymbol{u}}^h \cdot \nabla_{\boldsymbol{x}} \boldsymbol{u}^h \right) - \nabla_{\boldsymbol{x}} \cdot \boldsymbol{\sigma}^h \left(\boldsymbol{u}^h, p^h \right) - \boldsymbol{f}^h \right] d\Omega. \quad (5.13)$$

The expression after τ is the streamline upwind perturbation, followed by the residual of the momentum equations, see (2.24). For the meshfree shape functions, which may be constructed to have any desired order of continuity, the summation expression over elements $\sum_{\mathrm{e}=1}^{n_{\mathrm{el}}} \int_{\Omega_{\mathrm{F,e}}}$ may simply be replaced by $\int_{\Omega_{\mathrm{F}}}$, which holds for the following stabilizations as well.

5.2.2 Pressure-Stabilizing/Petrov-Galerkin (PSPG)

Considering stabilization of mixed problems as described in section 5.1.2, the PSPG stabilization is a common technique. It has been introduced for the stabilization of the Stokes equations [96] and incompressible Navier-Stokes equations [179]. There, the problem comes from the continuity condition, which is also called incompressibility constraint, underlining the mixed character of this formulation. To avoid unnecessary confusion related to an introduction of abstract universal mixed formulations, the PSPG method is described for the special case of Navier-Stokes equations only.

PSPG weak form of the incompressible NS eqts. The following PSPG stabilization terms are added to the weak form (3.17)

$$\ldots + \sum_{\mathrm{e}=1}^{n_{\mathrm{el}}} \int_{\Omega_{\mathrm{F,e}}} \tau \left(\frac{1}{\varrho_{\mathrm{F}}} \nabla_{\boldsymbol{x}} q^h \right) \cdot \left[\varrho_{\mathrm{F}} \left(\boldsymbol{u}^h_{,t} + \bar{\boldsymbol{u}}^h \cdot \nabla_{\boldsymbol{x}} \boldsymbol{u}^h \right) - \nabla_{\boldsymbol{x}} \cdot \boldsymbol{\sigma}^h \left(\boldsymbol{u}^h, p^h \right) - \boldsymbol{f}^h \right] d\Omega. \quad (5.14)$$

In mixed convection-dominated problems, such as the incompressible Navier-Stokes equations with high Reynolds-numbers and equal-order interpolations, SUPG *and* PSPG (called herein SUPG/PSPG) stabilization have to be applied to obtain satisfactory results. Then, (5.13) and (5.14) are added to (3.17). The PSPG stabilization parameter τ does not necessarily have to be identical with the SUPG stabilization parameter [179]. However, this is often the case in practice, which can be explained by the close analogy to the GLS stabilization, where one stabilization parameter results naturally.

5.2.3 Galerkin/Least-Squares (GLS)

The GLS stabilization can be interpreted as a generalization of the SUPG method and was motivated from mathematical analysis rather than the artificial diffusion aspect. In the GLS method, the operator over the test functions is the differential operator of the original problem. For a general weak form (5.9), the test functions are modified as

$$\boldsymbol{w}^\star = \boldsymbol{w} + \tau \mathcal{L} \boldsymbol{w}, \quad \boldsymbol{w} = \boldsymbol{N}, \tag{5.15}$$

where $\mathcal{L}$ is the whole operator of the PDE under consideration. For example, in case of the advection-diffusion equation (5.2), $\mathcal{L} = \boldsymbol{c} \cdot \nabla - \nabla \cdot (\mathbf{K}\ \nabla)$. One can thus see that the difference to the SUPG is in the modification of $\tau \mathcal{L} \boldsymbol{w}$ for the GLS instead of $\tau \mathcal{L}_{\mathrm{adv}} \boldsymbol{w}$ for the SUPG. For hyperbolic systems (no diffusion, i.e. second order terms) and/or linear test and shape functions, the GLS stabilization reduces to the SUPG stabilization [97].

GLS weak form of the incompressible NS eqts. Applying GLS stabilization to the incompressible Navier-Stokes equations leads the following terms added to (3.17)

$$\begin{aligned} \ldots + \sum_{\mathrm{e}=1}^{n_{\mathrm{el}}} \int_{\Omega_{\mathrm{F,e}}} \tau \frac{1}{\varrho_{\mathrm{F}}} \left[\varrho_{\mathrm{F}} \left(\bar{\boldsymbol{u}}^h \cdot \nabla_{\boldsymbol{x}} \boldsymbol{w}^h \right) - \nabla_{\boldsymbol{x}} \cdot \boldsymbol{\sigma}^h \left(\boldsymbol{w}^h, q^h \right) \right] \cdot \\ \left[\varrho_{\mathrm{F}} \left(\boldsymbol{u}^h_{,t} + \bar{\boldsymbol{u}}^h \cdot \nabla_{\boldsymbol{x}} \boldsymbol{u}^h \right) - \nabla_{\boldsymbol{x}} \cdot \boldsymbol{\sigma}^h \left(\boldsymbol{u}^h, p^h \right) - \boldsymbol{f}^h \right] d\Omega. \end{aligned} \tag{5.16}$$

The time derivative $\boldsymbol{w}^h_{,t}$ of the test functions in the perturbation is omitted here (as well as in the SUPG stabilization) because the time-discretization described in section 3.3 does not need stabilization in time direction. GLS stabilization allows arbitrary combinations of interpolations, which is realized by circumventing the Babuška-Brezzi conditions from the beginning, see e.g. [95]. Hence, oscillations and other problems described in sections 5.1.1 and 5.1.2 can be stabilized with GLS stabilization. It is an interesting fact that SUPG/PSPG stabilization for the incompressible Navier-Stokes equations can be motivated from the GLS stabilization with only a few reductions [179]; in case of linear FEM, they fully agree. This becomes obvious by comparing the perturbations of the test functions

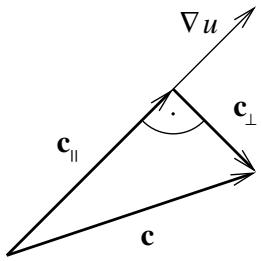

Figure 19: Projection of the advection direction $\boldsymbol{c}$ onto the solution gradient $\nabla \boldsymbol{u}$.

in both stabilizations:

$$\text{SUPG/PSPG} \; : \; \left[\bar{\boldsymbol{u}}^h \cdot \nabla_{\boldsymbol{x}} \boldsymbol{w}^h + \frac{1}{\varrho_{\mathrm{F}}} \nabla_{\boldsymbol{x}} \cdot q^h \right], \tag{5.17}$$

$$\text{GLS} \; : \; \left[\bar{\boldsymbol{u}}^h \cdot \nabla_{\boldsymbol{x}} \boldsymbol{w}^h + \frac{1}{\varrho_{\mathrm{F}}} \nabla_{\boldsymbol{x}} \cdot q^h - \frac{\mu}{\varrho_{\mathrm{F}}} \nabla_{\boldsymbol{x}} \cdot \left(\nabla_{\boldsymbol{x}} \boldsymbol{w}^h + \left(\nabla_{\boldsymbol{x}} \boldsymbol{w}^h \right)^{\mathrm{T}} \right) \right], \tag{5.18}$$

which shows that there are additional second order (diffusion) terms in the GLS.

5.2.4 Discontinuity Capturing

As pointed out in section 5.1.3 over- and undershoots in the solution can be prevented by getting control in the direction of the solution gradient. Using a Petrov-Galerkin approach, this is done with the following modification of the test functions [101]

$$\boldsymbol{w}^\star = \boldsymbol{w} + \tau \boldsymbol{c}_{\parallel} \cdot \boldsymbol{w}, \tag{5.19}$$

where $\tau \boldsymbol{c}_{\parallel} \cdot \boldsymbol{w}$ is the discontinuity-capturing term. In addition, a stabilization with SUPG or GLS is necessary to get control in the direction of the streamline. The parameter $\boldsymbol{c}_{\parallel}$ is a projection of the advection direction $\boldsymbol{c}$ onto the solution gradient $\nabla \boldsymbol{u}$ as shown in Fig. 19. It is defined as

$$\boldsymbol{c}_{\parallel} = \begin{cases} \frac{\boldsymbol{c} \cdot \nabla \boldsymbol{u}}{|\nabla \boldsymbol{u}|_2^2} \nabla \boldsymbol{u}, & \text{if } \nabla \boldsymbol{u} \neq 0 \\ 0, & \text{if } \nabla \boldsymbol{u} = 0 \end{cases} . \tag{5.20}$$

The parameter τ is defined differently from the stabilization parameters for SUPG, PSPG and GLS [101]. In the examples presented in this work, suitable results are obtained without explicitely smoothing over- and undershoots. Therefore, discontinuity capturing is no longer considered in what follows.

5.3 The Stabilization Parameter

Each of the stabilization methods described above consists of two ingredients: The structure of the perturbation and the stabilization parameter τ. It can easily be shown that the same arguments for the structure of the stabilization schemes hold both for meshfree and meshbased methods, e.g. [89]. However, this is in general not true for the stabilization parameter τ itself.

In the finite element context, there are several suggestions for the determination of τ in the literature, i.e. with the help of element matrix and vector norms [180], the Green's function of the element [93], mathematical error analysis [53, 54, 97], or model equations [42, 80, 110].

From mathematical analysis in the finite element context, one can find the following design criteria for the stabilization parameter: $\tau > 0$ in general, $\tau = O\left(h^2/K\right)$ for low element Peclet numbers $\mathrm{Pe}= |\boldsymbol{c}|\, h/\left(2K\right)$, and $\tau = O\left(h/|\boldsymbol{c}|\right)$ for high Peclet numbers, where h is a measure of the node distribution, and K and $|\boldsymbol{c}|$ are measures of the diffusion and convection respectively. A number of formulas that fulfill these basic requirements for the stabilization parameter are available in the finite element context, see section 5.3.1.

The question of an "optimal" stabilization parameter τ requires an optimality criterion of the resulting approximation. Often the one-dimensional advection-diffusion equation is taken as a model equation. There, the exact solution is known, and enables one to calculate stabilization parameters that fulfill any desired optimality criterion. An optimality criterion that has proven to be particularly useful is the one that obtains *the nodally exact solution* of the model equation. It can be shown that for linear FEM and a regular node distribution, the "coth-formula"

$$\tau = \frac{\Delta x}{2c}\left(\coth\left(\mathrm{Pe}\right) - \frac{1}{\mathrm{Pe}}\right), \quad \mathrm{Pe} = \frac{|\boldsymbol{c}|\, h}{2K}, \quad \Delta x = x_i - x_{i-1} = \mathrm{const} \tag{5.21}$$

fulfills this criterion and leads to nodally exact approximations. This formula has been generalized in a straightforward way to multi-dimensions, and is—together with similar versions—frequently used in practice for the successful stabilization of arbitrary problems with linear FEM; and this although it is derived only from the special case of the one-dimensional advection-diffusion equation. It has been shown in [53, 54] that straightforward use of this formula for higher-order FEM is not justified in general, and requires some modifications. It may thus be presumed that using these standard formulas derived in the

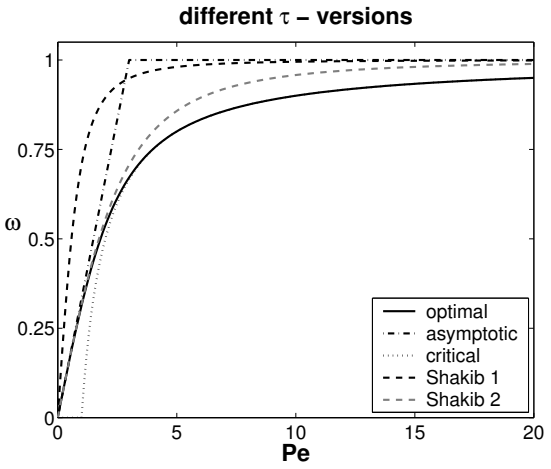

Figure 20: Alternative versions of the stabilization parameter $\tau = \frac{\Delta x}{2c}\omega$.

meshbased context of the linear FEM is also not suitable for MMs in general.

The standard way to obtain the coth-formula is to analytically solve the resulting *difference* equations in the system of equations emanating from the weak form of the model equation, discretized with linear FEM. Then, this solution is equated with the analytical solution of the *differential* equation [42, 80, 110]. An alternative way works with help of a Taylor series expansion [46]. In the following, another approach is presented, see also [60, 63]. This approach is found particularly useful to determine nodally exact solutions of the one-dimensional advection-diffusion equation with arbitrary (not only linear) finite element interpolations, and also with MMs. For a comparison of different possibilities to obtain formulas for τ see [60].

5.3.1 Standard Formulas for τ

In practice, alternative versions of the stabilization parameter are used instead of the "optimal" coth-version (5.21), which is due to the fact that they are less time-consuming to compute. They can be considered as approximations of the coth-formula and are compared in Fig. 20. Instead of

$$\tau = \frac{\Delta x}{2c}\left(\coth\left(\mathrm{Pe}\right) - \frac{1}{\mathrm{Pe}}\right) = \frac{\Delta x}{2c}\omega, \tag{5.22}$$

only ω ("diffusion correction factor" [168]) is visualized as a function of the element Peclet number.

- optimal version, first in [42]

$$\omega = \left(\coth\left(\mathrm{Pe}\right) - \frac{1}{\mathrm{Pe}} \right) \tag{5.23}$$

- doubly asymptotic approximation [31, 91]

$$\omega = \begin{cases} \mathrm{Pe}/3, & -3 \leq \mathrm{Pe} \leq 3 \\ \mathrm{sgn}\left(\mathrm{Pe}\right), & \mathrm{Pe} > 3 \end{cases} \tag{5.24}$$

- critical approximation [31, 42, 91]

$$\omega = \begin{cases} -1 - 1/\mathrm{Pe}, & \mathrm{Pe} < 1 \\ 0, & -1 \leq \mathrm{Pe} \leq 1 \\ 1 - 1/\mathrm{Pe}, & \mathrm{Pe} > 1 \end{cases} \tag{5.25}$$

- versions of Shakib [168]

$$\omega_1 = \left(1 + \frac{1}{\mathrm{Pe}^2} \right)^{-1/2}, \; \omega_2 = \left(1 + \frac{9}{\mathrm{Pe}^2} \right)^{-1/2} \tag{5.26}$$

The first version of Shakib, ω_1, is maybe the one most often used in practice:

$$\tau = \frac{\Delta x}{2c}\omega_1 = \frac{\Delta x}{2c}\left(1 + \frac{1}{\mathrm{Pe}^2} \right)^{-1/2} = \left[\left(\frac{2c}{\Delta x} \right)^2 + \left(\frac{4K}{\Delta x^2} \right)^2 \right]^{-1/2}. \tag{5.27}$$

The two terms in the right expression can be interpreted as the advection-dominated and diffusion-dominated limit [180]. It can be seen that the dependency on the mesh size Δx in the advection-dominated case is $\tau \approx \Delta x / (2c)$, hence $O(\Delta x)$, while it is in the diffusion-dominated case $\tau \approx \Delta x^2 / (4K)$, hence $O(\Delta x^2)$. In multi-dimensions the parameter Δx is replaced by suitable length measures, in the finite element context in general by the element length h_e. For instationary problems an additional time term of $(2/\Delta t)^2$ is added in (5.27).

5.3.2 One-Dimensional Advection-Diffusion Equation

The strong form of the one-dimensional advection-diffusion equation is

$$c\frac{\partial u\left(x\right)}{\partial x}-K\frac{\partial^{2}u\left(x\right)}{\partial x^{2}}=0,\quad x\in\Omega,\quad c,K\in\mathbb{R} \tag{5.28}$$

with suitable boundary conditions. A scalar quantity $u\left(x\right)$ is advected with the velocity c and, in addition, experiences a diffusion depending on K. The exact solution of this problem is known as

$$u^{(\mathrm{ex})}=C_{1}e^{\gamma\cdot x}+C_{2},\quad \gamma=\frac{c}{K}. \tag{5.29}$$

MLS nodes are introduced at the positions $x_1, x_2, \ldots x_r$ inside the domain. Discretization of the SUPG stabilized weak form with $u^{h}\left(x\right)=\boldsymbol{N}^{\mathrm{T}}\left(x\right)\boldsymbol{u}$ gives

$$\int_{\Omega}\left(\boldsymbol{w}+\tau c\cdot\partial_{x}\boldsymbol{w}\right)\left(c\cdot\partial_{x}u^{h}-K\partial_{x}^{2}u^{h}\right)d\Omega=\boldsymbol{0}, \tag{5.30}$$

where $\partial_x = \partial/\partial x$.

One equation—say equation no. ℓ—is extracted of this system of equations,

$$\left[\int_{\Omega}\left(w_{\ell}+\tau_{\ell}c\cdot\partial_{x}w_{\ell}\right)\left(c\cdot\partial_{x}\boldsymbol{N}^{\mathrm{T}}-K\partial_{x}^{2}\boldsymbol{N}^{\mathrm{T}}\right)d\Omega\right]\boldsymbol{u}=\boldsymbol{0}. \tag{5.31}$$

This equation corresponds to node ℓ at x_ℓ with the test function w_ℓ. There is one τ_ℓ for each equation/node. Consequently, one may call this stabilization *nodal stabilization*, in contrast to element stabilization—where stabilization parameters τ_{e} for each element matrix are used—, which is standard in the FEM. See [60] for a detailed comparison of nodal and element stabilization.

The τ_ℓ-values of each equation are computed such that the nodally exact solution is obtained. This can be done by introducing the exact solution into the vector $\boldsymbol{u}$. One has $u^{(\mathrm{ex})}\left(x_j\right)=u_j^{(\mathrm{ex})}=C_1e^{\gamma\cdot x_j}+C_2$, and according to the ansatz $u^{h}\left(x_j\right)=u_j^{h}=\sum N_i\left(x_j\right)u_i$. Nodal exactness means

$$u_j^{h} = u_j^{(\mathrm{ex})}, \tag{5.32}$$

$$\sum N_i\left(x_j\right)u_i = C_1e^{\gamma\cdot x_j}+C_2, \tag{5.33}$$

$$\boldsymbol{D}\boldsymbol{u} = \boldsymbol{u}^{(\mathrm{ex})}, \tag{5.34}$$

where $\boldsymbol{D} = D_{ij} = N_i(x_j)$ is a $n \times n$ matrix of the n shape functions evaluated at the n nodal positions. $\boldsymbol{D}$ is a sparse matrix if the shape functions are non-zero only in small parts of the domain Ω. In the FEM, the shape functions have local supports, specified indirectly with help of the mesh, whereas the supports of MMs are defined with help of the dilatation parameter ρ, see section 4.3.3. For shape functions with Kronecker-δ property, $N_i(x_j) = \delta_{ij}$ and thus $\boldsymbol{D} = \boldsymbol{I}$.

Rearranging (5.31) for τ_ℓ and replacing $\boldsymbol{u}$ with $\boldsymbol{D}^{-1}\boldsymbol{u}^{(\mathrm{ex})}$ results in

$$\tau_\ell = -\frac{\left[\int_\Omega (w_\ell)\left(c\cdot\partial_x\boldsymbol{N}^{\mathrm{T}} - K\partial_x^2\boldsymbol{N}^{\mathrm{T}}\right)d\Omega\right]\boldsymbol{D}^{-1}\boldsymbol{u}^{(\mathrm{ex})}}{\left[\int_\Omega (c\cdot\partial_x w_\ell)\left(c\cdot\partial_x\boldsymbol{N}^{\mathrm{T}} - K\partial_x^2\boldsymbol{N}^{\mathrm{T}}\right)d\Omega\right]\boldsymbol{D}^{-1}\boldsymbol{u}^{(\mathrm{ex})}} \tag{5.35}$$

$$= -\frac{\left[\int_\Omega (w_\ell)\left(c\cdot\partial_x\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1} - K\partial_x^2\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1}\right)d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}}{\left[\int_\Omega (c\cdot\partial_x w_\ell)\left(c\cdot\partial_x\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1} - K\partial_x^2\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1}\right)d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}}. \tag{5.36}$$

This expression for τ_ℓ leads to nodally exact solutions for arbitrary shape and test functions and arbitrary point distributions. In what follows, this result will be interpreted.

5.3.3 Linear FEM

In the case of linear finite element shape functions, a number of simplifications for (5.36) is possible. Due to the Kronecker-δ property of the nodal finite element shape functions, one finds $\boldsymbol{D} = \boldsymbol{D}^{-1} = \boldsymbol{I}$. Partial integration is applied to the diffusion term in the nominator, whereas this term cancels out in the denominator (assuming that the second derivatives of the linear finite element shape functions are 0 everywhere in Ω). It remains for τ_ℓ (for constant c and K):

$$\tau_\ell = -\frac{\left[c\int_\Omega w_\ell\partial_x\boldsymbol{N}^{\mathrm{T}}d\Omega + K\int_\Omega \partial_x w_\ell\partial_x\boldsymbol{N}^{\mathrm{T}}d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}}{\left[c^2\int_\Omega \partial_x w_\ell\partial_x\boldsymbol{N}^{\mathrm{T}}d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}}, \tag{5.37}$$

$$= -\frac{\left[c\int_\Omega w_\ell\partial_x\boldsymbol{N}^{\mathrm{T}}d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}}{\left[c^2\int_\Omega \partial_x w_\ell\partial_x\boldsymbol{N}^{\mathrm{T}}d\Omega\right]\boldsymbol{u}^{(\mathrm{ex})}} - \frac{K}{c^2}. \tag{5.38}$$

The integral expressions in (5.38) can be evaluated explicitely for the case of linear shape and test functions and a regular node distribution as

$$\int_\Omega w_\ell\partial_x\boldsymbol{N}^{\mathrm{T}}d\Omega = \frac{1}{2}\left[\ -1,\ \ 0,\ \ 1\ \right], \tag{5.39}$$

$$\int_\Omega \partial_x w_\ell\partial_x\boldsymbol{N}^{\mathrm{T}}d\Omega = \frac{1}{\Delta x}\left[\ -1,\ \ 2,\ \ -1\ \right]. \tag{5.40}$$

The scalar product of these expressions with $\boldsymbol{u}^{(\mathrm{ex})} = C_1 e^{\gamma \cdot \boldsymbol{x}} + C_2$ gives

$$\tau_\ell = \frac{\Delta x}{2c} \frac{E_{\ell+1} - E_{\ell-1}}{E_{\ell-1} + 2E_\ell + E_{\ell+1}} - \frac{K}{c^2}, \tag{5.41}$$

$$= \frac{\Delta x}{2c} \frac{\sinh(\gamma \Delta x)}{\cosh(\gamma \Delta x) - 1} - \frac{K}{c^2}, \tag{5.42}$$

$$= \frac{\Delta x}{2c} \left(\coth(\mathrm{Pe}) - \frac{1}{\mathrm{Pe}} \right), \tag{5.43}$$

with $E_j = C_1 e^{\gamma \cdot x_j} + C_2$ and $\mathrm{Pe} = \gamma \cdot \Delta x/2 = c \cdot \Delta x/(2K)$. With this definition of the stabilization parameter one obtains the nodally exact solution for the one-dimensional advection-diffusion equation, approximated with linear FEM and a regular node distribution.

Using standard element stabilization instead of nodal stabilization with $\tau_\mathrm{e} = \tau_\ell$ leads to the same result. This formula for τ has often been called "optimal" in the literature, e.g. in [31, 42, 80, 110]. It has a *local* character as it is independent of the boundary conditions and only relies on the relative positions of the neighbouring nodes $x_{\ell-1}$ and $x_{\ell+1}$. Although it is derived for the very special case of the one-dimensional advection-diffusion equation approximated with linear finite elements and a regular node distribution, it has been generalized in a straightforward way to instationary multidimensional advection-dominated problems approximated with linear FEM in arbitrary nodal arrangements, see e.g. [60] for an explanation.

5.3.4 Quadratic FEM

In this section it is briefly shown that the same procedure may be applied to obtain optimal stabilization parameters τ—leading to nodally exact solutions—with quadratic elements (and any other). Again, a regular node distribution is assumed. Starting point is (5.36), which for shape functions with Kronecker-δ property is

$$\tau_\ell = -\frac{\left[c \int_\Omega w_\ell \partial_x \boldsymbol{N}^\mathrm{T} d\Omega + K \int_\Omega \partial_x w_\ell \partial_x \boldsymbol{N}^\mathrm{T} d\Omega\right] \boldsymbol{u}^{(\mathrm{ex})}}{\left[c^2 \int_\Omega \partial_x w_\ell \partial_x \boldsymbol{N}^\mathrm{T} d\Omega - cK \int_\Omega \partial_x w_\ell \partial_x^2 \boldsymbol{N}^\mathrm{T}\right] \boldsymbol{u}^{(\mathrm{ex})}}. \tag{5.44}$$

For quadratic elements the system of equations has *two* different difference equations instead of only one for linear elements, see Fig. 21. This is because there are 3×3 element matrices and there is one difference equation I_a which has a five node stencil and another

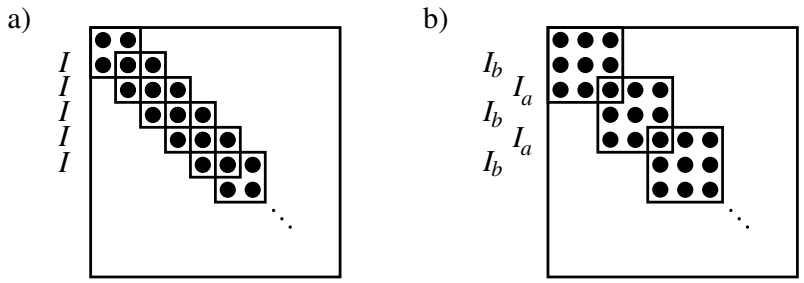

Figure 21: Assembly of element matrices from quadratic elements results in two different difference stencils I_a and I_b, whereas linear elements only lead to one difference stencil I.

I_b with gives a three node stencil only. Clearly, each of the two equations requires an individual τ.

Evaluating (5.44) for equation I_a and inserting the exact solution evaluated at the nodes $\boldsymbol{u}^{(\mathrm{ex})}$ leads to

$$\tau_{I_a} = -\frac{\left(c\left[\frac{1}{6},-\frac{2}{3},0,\frac{2}{3},-\frac{1}{6}\right]+\frac{K}{\Delta x}\left[\frac{1}{3},-\frac{8}{3},\frac{14}{3},-\frac{8}{3},\frac{1}{3}\right]\right)\boldsymbol{u}^{(\mathrm{ex})}}{\left(\frac{c^2}{\Delta x}\left[\frac{1}{3},-\frac{8}{3},\frac{14}{3},-\frac{8}{3},\frac{1}{3}\right]-\frac{cK}{\Delta x^2}[4,-8,0,8,-4]\right)\boldsymbol{u}^{(\mathrm{ex})}} \tag{5.45}$$

$$= \frac{\Delta x}{2c}\frac{\frac{2}{3}\sinh(2\mathrm{Pe})-\frac{8}{3}\sinh(\mathrm{Pe})-\frac{1}{\mathrm{Pe}}\left[\frac{2}{3}\cosh(2\mathrm{Pe})-\frac{16}{3}\cosh(\mathrm{Pe})+\frac{14}{3}\right]}{\frac{2}{3}\cosh(2\mathrm{Pe})-\frac{16}{3}\cosh(\mathrm{Pe})+\frac{14}{3}-\frac{1}{\mathrm{Pe}}\left[-4\sinh(2\mathrm{Pe})+8\sinh(\mathrm{Pe})\right]}. \tag{5.46}$$

The same can be done for equation I_b

$$\tau_{I_b} = -\frac{\left(c\left[-\frac{2}{3},0,\frac{2}{3}\right]+\frac{K}{\Delta x}\left[-\frac{8}{3},\frac{16}{3},-\frac{8}{3}\right]\right)\boldsymbol{u}^{(\mathrm{ex})}}{\left(\frac{c^2}{\Delta x}\left[-\frac{8}{3},\frac{16}{3},-\frac{8}{3}\right]-\frac{cK}{\Delta x^2}[0,0,0]\right)\boldsymbol{u}^{(\mathrm{ex})}} \tag{5.47}$$

$$= \frac{\Delta x}{2c}\frac{-\frac{8}{3}\sinh(\mathrm{Pe})-\frac{1}{Pe}\left[-\frac{16}{3}\cosh(\mathrm{Pe})+\frac{16}{3}\right]}{-\frac{16}{3}\cosh(\mathrm{Pe})+\frac{16}{3}}. \tag{5.48}$$

Applying these two τ-definitions leads to nodally exact solutions as can be seen from the left part of Fig. 22. In the right part, the two definitions are compared with the coth-version for linear FEM. Most importantly, it is found that there are two different limits of τ_{I_a} and τ_{I_b}. Consequently, choosing only *one* τ for the stabilization seems inadequate.

Some conclusions for element stabilization with τ_{e} are possible. Having one τ_{e} for each element matrix does also not consider the two different limits for $\mathrm{Pe} \longrightarrow \infty$ of the two different types I_a and I_b of equations. However, in practice, this is still standard, see e.g. [168], where it is pointed out that for quadratic elements τ_{e} may be multiplied by one half. Looking at the two limits in Fig. 22, it becomes clear, why this particular value may

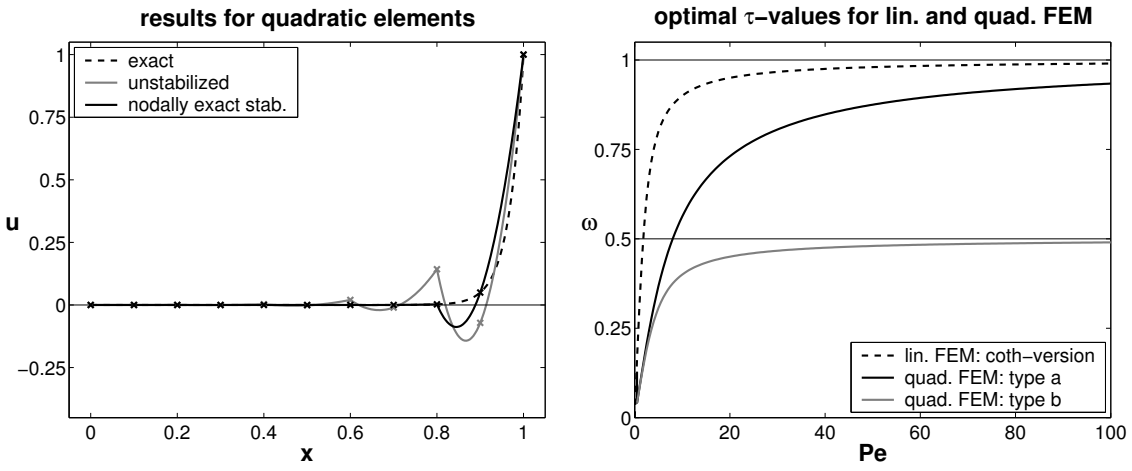

Figure 22: a) Nodally exact results with quadratic elements. b) Comparison of the different versions of $\tau = \frac{\Delta x}{2c}\omega$ for linear and quadratic elements.

be chosen. However, a treatment of the element equations as

$$\text{standard: } \tau_{\mathrm{e}} \begin{pmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{pmatrix} \qquad \text{proposed: } \begin{pmatrix} \tau_{\mathrm{e}_a} \begin{pmatrix} \times & \times & \times \end{pmatrix} \\ \tau_{\mathrm{e}_b} \begin{pmatrix} \times & \times & \times \end{pmatrix} \\ \tau_{\mathrm{e}_a} \begin{pmatrix} \times & \times & \times \end{pmatrix} \end{pmatrix}$$

seems more adequate, because then the different limits can be considered respectively. The advantage of this proposal can also be verified with numerical experiments. τ_{e_a} and τ_{e_b} are chosen equivalently to τ_{I_a} and τ_{I_b} by replacing Δx and Pe with the corresponding element numbers. This gives in case of a regular node distribution also for element stabilization nodally exact values. A straightforward simplification of (5.46) and (5.48) is to choose the coth-formula (5.43) for τ_{e_a} and $\tau_{\mathrm{e}_b} = \frac{1}{2}\tau_{\mathrm{e}_a}$.

This procedure for determining "optimal" formulas for τ may also be applied analogously to higher order elements, see [60]. Thereby nodally exact solutions may be also found for higher order approximations. In these cases individual formulas for τ for each line of the element matrix are obtained, in any case *local* formulas for τ result. Using standard formulas for τ derived for linear shape functions may not be well-suited here, see also [53, 54].

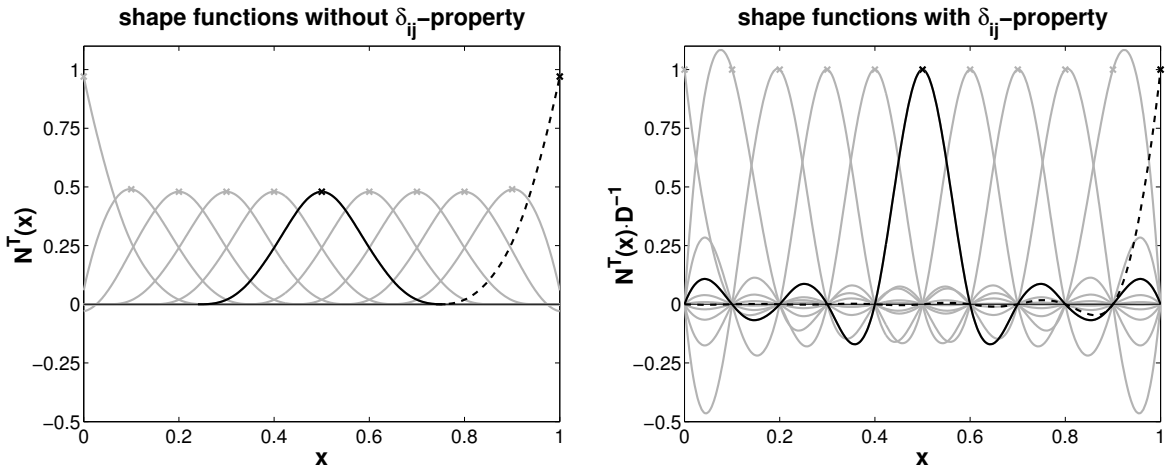

Figure 23: Local shape functions $\boldsymbol{N}^{\mathrm{T}}$ without Kronecker-δ property and transformed global shape functions $\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1}$ with Kronecker-δ property.

5.3.5 Meshfree Methods

For meshfree methods, (5.36) can not be simplified in general. This result is interpreted as follows. The expression for τ_ℓ in (5.36) is rewritten as

$$\tau_\ell = -\frac{\left[\int_\Omega f_1\left(w_\ell\right) g\left(\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1}\right) d\Omega\right] \boldsymbol{u}^{(\mathrm{ex})}}{\left[\int_\Omega f_2\left(w_\ell\right) g\left(\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1}\right) d\Omega\right] \boldsymbol{u}^{(\mathrm{ex})}}, \tag{5.49}$$

where f_1, f_2 and g are linear functions of the test and shape functions respectively. The expressions in the nominator and denominator are scalar products

$$\int_\Omega \underbrace{f_i\left(w_\ell\right)}_{1\times 1} \underbrace{g\left(\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1}\right)}_{1\times d} d\Omega \underbrace{\boldsymbol{u}^{(\mathrm{ex})}}_{d\times 1}. \tag{5.50}$$

The meshfree test and shape functions $\boldsymbol{w}$ and $\boldsymbol{N}$ have local supports. However, the term $\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1}$ can be interpreted as the *globalized* meshfree shape functions having Kronecker-δ property. This may be gleaned from Fig. 23, where local shape functions $\boldsymbol{N}^{\mathrm{T}}$ without and transformed global shape functions $\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1}$ with Kronecker-δ property are shown.

Consequently, the vectors $\int_\Omega f_i\left(w_\ell\right) g\left(\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1}\right) d\Omega$ are full vectors, which is in contrast to shape functions having Kronecker-δ property. In the latter case, $g\left(\boldsymbol{N}^{\mathrm{T}}\boldsymbol{D}^{-1}\right) = g\left(\boldsymbol{N}^{\mathrm{T}}\right)$, and the vector is sparse. Evaluating the scalar product with $\boldsymbol{u}^{(\mathrm{ex})}$ shows the important difference: Shape functions without Kronecker-δ property have non-zero entries in the

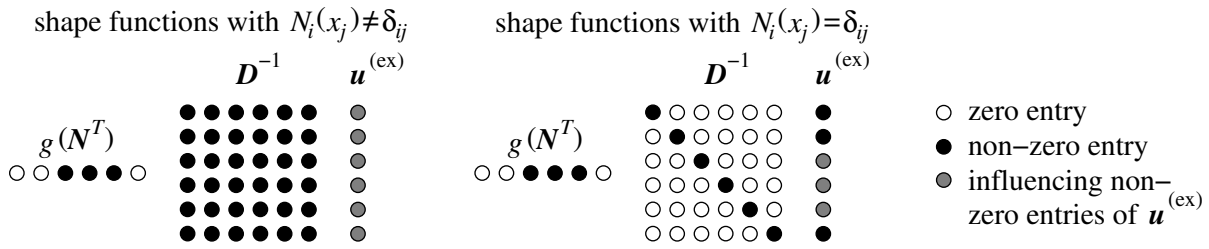

Figure 24: Evaluating the scalar products for τ_ℓ for shape function without and with Kronecker-δ property, in the former case all entries of $\boldsymbol{u}^{(\mathrm{ex})}$ have influence in the result.

scalar-product for *all* components of the vector $\boldsymbol{u}^{(\mathrm{ex})}$, whereas, in contrast, shape functions with Kronecker-δ property only have non-zero entries for the *neighbouring* nodes. This may be seen symbolically from Fig. 24, where it is clear that the nodally exact τ_ℓ for shape functions without Kronecker-δ property can only be obtained with a global criterion, because all entries of $\boldsymbol{u}^{(\mathrm{ex})}$ have an influence on the result.

Keeping in mind that $\boldsymbol{u}^{(\mathrm{ex})}$ is an exponential function, the scalar product will depend more and more on the last entry of this vector as the convection-diffusion ratio $\gamma = c/K$ grows, because then

$$u^{(\mathrm{ex})}(x_n) = u_n^{(\mathrm{ex})} \gg u_i^{(\mathrm{ex})} = u^{(\mathrm{ex})}(x_i) \quad \forall i \neq n. \tag{5.51}$$

The last component of $\boldsymbol{u}^{(\mathrm{ex})}$ is $u_n^{(\mathrm{ex})}$, and belongs to node n with the largest x-value, i.e. the global downstream node. The conclusion is that the stabilization parameter τ_ℓ, leading to nodally exact solutions has a *global* character, as it depends on all node positions and for convection-dominated cases most importantly on the global downstream node. This is in contrast to shape functions with Kronecker-δ property, whose stabilization relies on the neighbouring nodes only. Therefore, it can in general not be expected that using the simple coth-formula—or other alternative similar versions derived as a local stabilization criterion for linear FEM—is successful also for MMs.

5.3.6 Small Dilatation Parameters

Meshfree shape functions are constructed with help of the node distribution and the definition of supports, see section 4. The support sizes are defined by the dilatation parameter ρ. It is a well known fact that MLS shape functions in one dimension with first order consistency become more and more equal to the standard nodal linear shape functions of the FEM as the dilatation parameter ρ approaches Δx. This is also shown in Fig. 25.

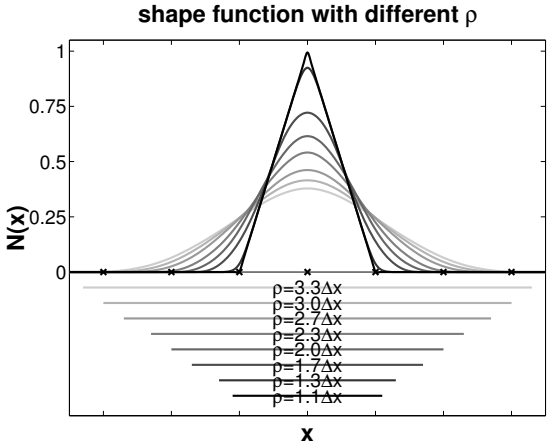

Figure 25: Meshfree shape function in a regular node distribution with varying dilatation parameter ρ.

Hence, it may be concluded that when $\rho \longrightarrow \Delta x$, the coth-formula becomes more and more suited also for MMs. Hence

$$\rho \longrightarrow 1 \cdot \Delta x \quad : \quad \boldsymbol{N}^{(\mathrm{MM})} \longrightarrow \boldsymbol{N}^{(\text{lin. FEM})} \tag{5.52}$$

$$\Rightarrow \quad \tau_\ell^{(\mathrm{MM})} \longrightarrow \tau_\ell^{(\text{lin. FEM})} = \frac{\Delta x}{2c}\left(\coth\left(\mathrm{Pe}\right) - \frac{1}{\mathrm{Pe}}\right). \tag{5.53}$$

A stability criterion of the MLS requires $\rho > \Delta x$ for linear consistency [137]. Thus, one can never reach the limit $\rho = \Delta x$, where the coth-formula gives the nodally exact solution. However, the proposal is that for reasonable advection-diffusion ratios and "small" dilatation parameters a successful stabilization with standard formulas—derived for meshbased methods—can be obtained. Dilatation parameters of $1.3\Delta x \leq \rho \leq 1.7\Delta x$ are suggested. For smaller ρ, the condition number of the MLS system of equations which has to be solved at every integration point may be too large to allow a sufficiently accurate solution, and for larger ρ the stabilization may not be reliable. The numerical results in section 5.4 confirm this assumption.

5.3.7 Stabilization Parameter in Multi-Dimensions

In the FEM, i.e. in the meshbased context, the generalization of the τ-formulas derived from the one-dimensional advection-diffusion equation to multi-dimensions is straightforward [31]. The one-dimensional parameters Δx and c are replaced with the element length

h_e and the norm of the advection $|\boldsymbol{c}|$. Assuming small dilatation parameters, the same generalization is proposed for meshfree methods. Hence τ_ℓ in multi-dimensions may be computed with

$$\tau_\ell = \frac{h_\rho}{2\,|\boldsymbol{c}|}\left(\coth\left(\mathrm{Pe}_\rho\right) - \frac{1}{\mathrm{Pe}_\rho}\right) \quad \text{with } \mathrm{Pe}_\rho = \frac{|\boldsymbol{c}|\,h_\rho}{2K} \tag{5.54}$$

or any other of the alternative versions for τ, see section 5.3.1. Here h_ρ is the "support length", analogously to the "element length" h_e in the meshbased context.

- min-version:

$$h_\rho = \min\left(\rho_x, \rho_y\right) \tag{5.55}$$

- max-version:

$$h_\rho = \max\left(\rho_x, \rho_y\right) \tag{5.56}$$

- inner-ellipsoid-version:

$$h_\rho = \sqrt{\frac{\left(1 + \frac{c_y^2}{c_x^2}\right)\cdot \rho_y^2}{\left(\frac{c_y}{c_x}\right)^2 + \left(\frac{\rho_y}{\rho_x}\right)^2}} \tag{5.57}$$

- real-length-version:

$$h_\rho = \min\left(\frac{\rho_x}{|c_x|}, \frac{\rho_y}{|c_y|}\right)\cdot\sqrt{c_x^2 + c_y^2} \tag{5.58}$$

Fig. 26 shows several possibilities to interpret h_ρ in case of rectangular supports. The support lengths for circular and ellipsoid supports can be directly read of from these formulas.

In case of the incompressible Navier-Stokes equations in two dimensions, the advection coefficients $\boldsymbol{c} = (c_x, c_y)$ are replaced by the convective velocities $\boldsymbol{u} = (u, v)$. In case of the ALE formulated Navier-Stokes equations [98], the convective velocity is $\bar{\boldsymbol{u}} = \boldsymbol{u} - \boldsymbol{u}_\mathrm{M}$, where $\boldsymbol{u}_\mathrm{M}$ is the mesh (or node) velocity. In the numerical experiments it is found that particularly the min-version (5.55) works very successfully also for large aspect ratios ($\rho_x/\rho_y \gg 1$, or $\rho_y/\rho_x \gg 1$). See [146] for an interesting parallel for high-aspect *elements*: There it is found that the minimal edge length works better than other versions for h_e.

The inner-ellipsoid-version (5.57) and the real-length-version (5.58) are dependent of the streamline direction of the flow inside the support. In case of the incompressible Navier-Stokes equations, this introduces some disadvantages: A representative streamline direction has to be found for the whole support, the streamline direction changes with each iteration

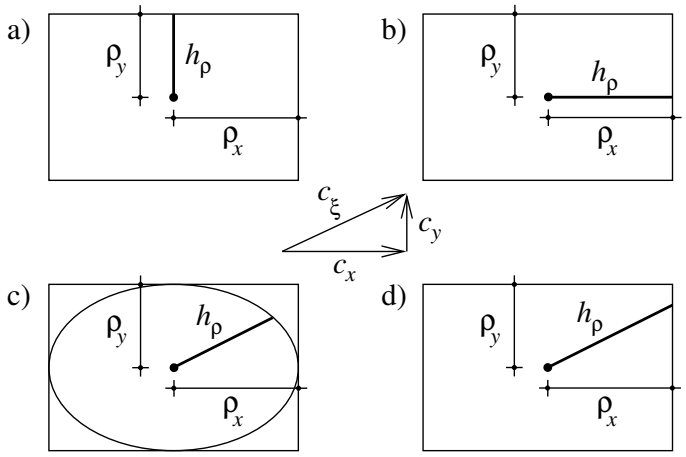

Figure 26: Different versions to compute the support length in streamline direction; a) min-version, b) max-version, c) inner-ellipsoid-version, d) real-length-version.

and/or time step, and the non-linearity introduced by $\tau = f(u, v, h_\rho)$ is more complex as compared with the min-version (5.55) and max-version (5.56).

5.4 Numerical Results

Numerical results are shown for two different problems, which are approximated by meshfree shape functions only. The first is the one-dimensional advection-diffusion equation (5.28), the model equation of section 5.3.2. It is shown that nodally exact solutions with meshfree shape functions may be obtained, but therefore, the *global* stabilization criterion (5.36) is needed. Furthermore, it is shown that standard formulas for τ are successful only for small dilatation parameters, which confirms the assumption from section 5.3.6 due to (5.53).

The second problem are the stationary incompressible Navier-Stokes equations in two dimensions. SUPG/PSPG and GLS stabilization according to (5.13), (5.14), and (5.16) are applied and compared. Small dilatation parameters are a crucial ingredient to obtain successfully stabilized results. Using supports with too large dilatation parameters results in degradation of convergence and solutions that are still either too oscillatory or too diffusive. The intention is to show that stabilization with small dilatation parameters

- smoothes out oscillations successfully

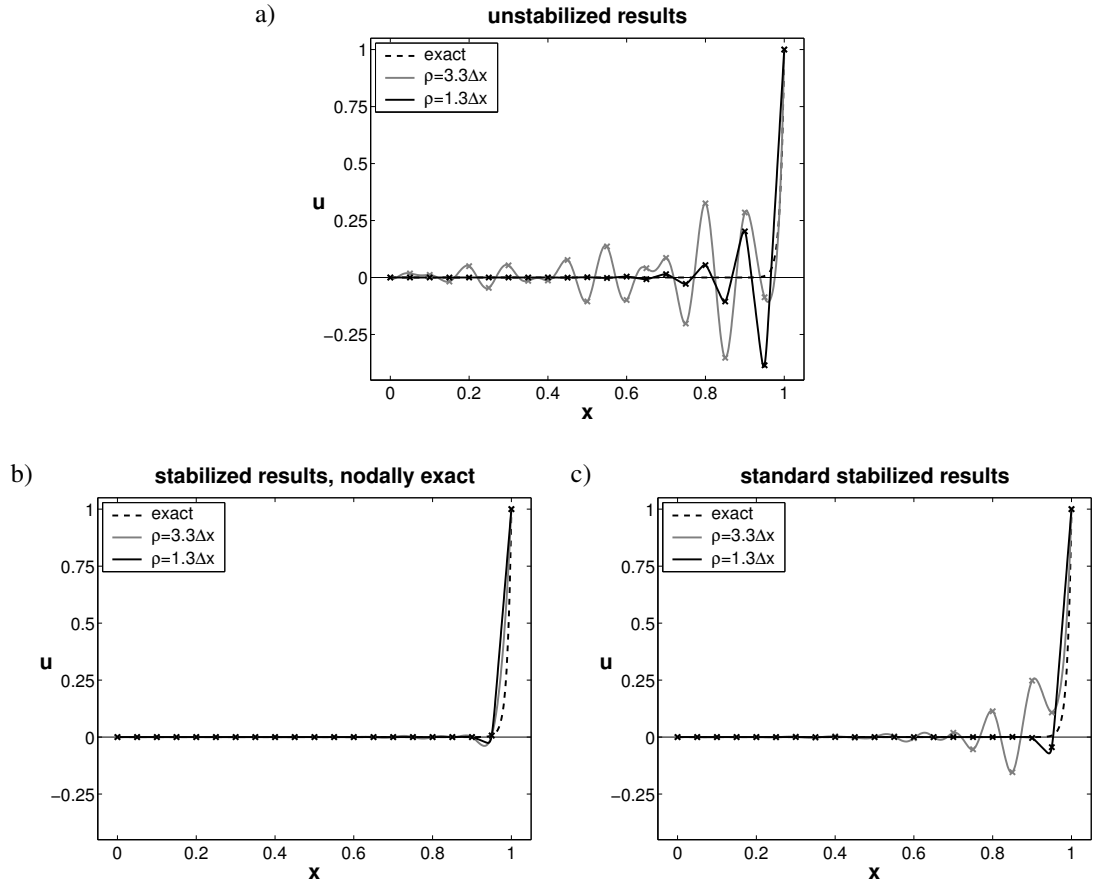

Figure 27: Results for the 1D advection-diffusion equation; a) without any stabilization, b) with the global stabilization criterion (5.36), c) with the local coth-formula (5.54).

- does not degrade accuracy in cases where stabilization is not necessary
- works also for anisotropic supports.

5.4.1 One-dimensional Advection-Diffusion Equation

The one-dimensional advection diffusion equation (5.28) is solved with 21 MLS nodes. The advection-diffusion ratio is $\gamma = c/K = 100$. Fig. 27a) shows the unstabilized results for two different dilatation parameters $\rho = 1.3\Delta x$ ("small") and $\rho = 3.3\Delta x$ ("large"). It can be seen that higher dilatation parameters lead to more oscillations, simply due to their higher Peclet number, see (5.54). Clearly, for both cases, stabilization is required.

Fig. 27b) shows the nodally exact result, which can be obtained with the global stabilization

criterion for τ_ℓ, see (5.36). In Fig. 27c) it can be seen that standard formulas for τ_ℓ like the coth-formula (5.54) only lead to successful stabilization when the dilatation parameter is small. Comparison of Fig. 27b) and c) shows that for small dilatation parameters, the result of the complicated global criterion (5.36) and the coth-criterion (5.54) gives almost the same result. This, however, is not the case for the large dilatation parameter of $\rho = 3.3\Delta x$, where pronounced oscillations remain in the solution. These oscillations are clearly not a problem of the high gradient itself that could not be captured by shape functions with such a large dilatation parameter (then the result of Fig. 27b) should have been oscillatory, too, and this is not the case), but results from the use of unsuitable stabilization parameters.

The conclusion is that the assumption of section 5.3.6 is confirmed: Standard formulas of meshbased methods for the stabilization parameter τ can only be reliably used for MMs with small dilatation parameters.

5.4.2 Driven Cavity Flow

The following test cases solve the stationary incompressible Navier-Stokes equations. The driven cavity problem is a standard test case with reference solutions given in [67] for a variety of Reynolds numbers. A flow inside a quadratic domain $\Omega = (0,1) \times (0,1)$ with no-slip boundary conditions on the left, right and lower wall develops under a shear flow of $u = 1.0$ and $v = 0.0$ applied on the upper boundary until a stationary solution is reached. Herein, this problem is solved with a fluid density of $\varrho_{\mathrm{F}} = 1.0$ and a viscosity of $\mu = 0.001$, leading to a Reynolds number of $\mathrm{Re} = \varrho_{\mathrm{F}} \cdot u \cdot L/\mu = 1000$. For a problem statement see Fig. 28, showing also streamlines and pressure distribution for $\mathrm{Re} = 1000$. In the sequel, only *velocity profiles* are studied at certain cuts through the two-dimensional domain, and linear interpolation is applied in-between the nodes for simplicity.

The first results are produced with 21×21 regularly distributed MLS nodes. Fig. 29 shows velocity profiles for u and v at $y = 0.95$, i.e. near the tangential flow boundary, where most of the oscillations occur. Two different dilatation parameters are shown, $\rho = 1.3\Delta x$ and $\rho = 2.3\Delta x$. Dilatation parameters $\rho > 2.7\Delta x$ converged either not at all or only very badly, underlining the need for small dilatation parameters, when standard formulas for τ_ℓ are used.

One can clearly see that the oscillations apparent in the unstabilized result are smoothed out successfully, especially for the case where $\rho = 1.3\Delta x$. For $\rho = 2.3\Delta x$ one may see

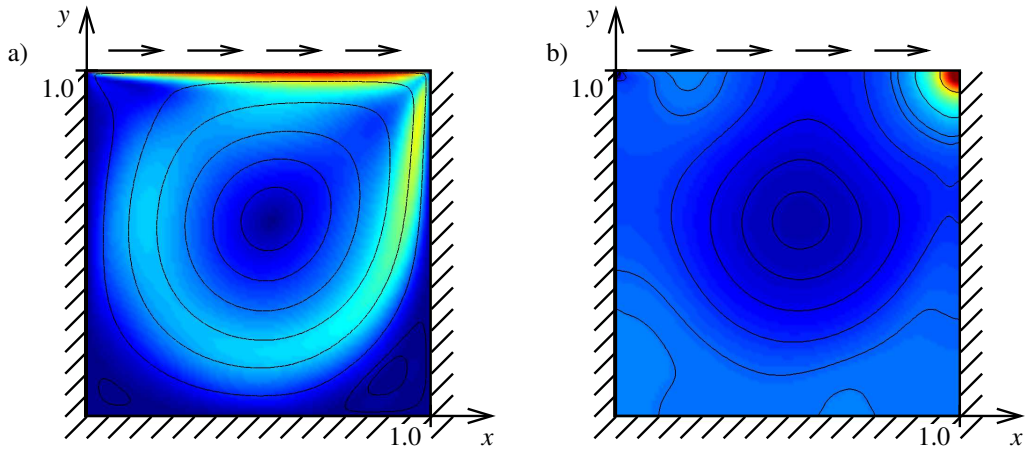

Figure 28: Problem statement of the driven cavity test case. As an example the a) velocity magnitude and b) pressure field is shown at Re = 1000.

from the velocity profile for v that very slight oscillations remain in this case. Again, the assumption that shape functions with small dilatation parameters can be stabilized very successfully is confirmed.

Fig. 30 shows the center velocity profiles for the case where $\rho = 1.3\Delta x$. Although along these cuts no oscillations are apparent in the unstabilized case, the stabilized profiles give better results. The reason for this is that the oscillations in the unstabilized case near the tangential flow boundary degrade the overall solution. Obviously, stabilization for shape functions with small dilatation parameters smoothes out oscillations successfully and leads to superior overall solutions in comparison to unstabilized calculations. It should be mentioned that the rather big difference to the reference solution given in [67] is due to the coarse node distribution and improves clearly for more refined distributions as shown later.

The next results are computed with 101×101 MLS nodes and $\rho = 1.3\Delta x$. With this large number of nodes, stabilization is not needed at all, i.e. the unstabilized solution is already free of oscillations. The results show that stabilization does not degrade the accuracy when it is not needed. Fig. 31 shows the center velocity profiles. It is interesting that unstabilized and SUPG/PSPG stabilized results are indistinguishable, whereas GLS stabilized results are slightly more diffusive. This was confirmed in a number of additional computations.

Fig. 32 shows a comparison of the reference solution with the meshfree solution (with $\rho = 1.3\Delta x$) and the solution from the P1/P1 triangular element with the same number of

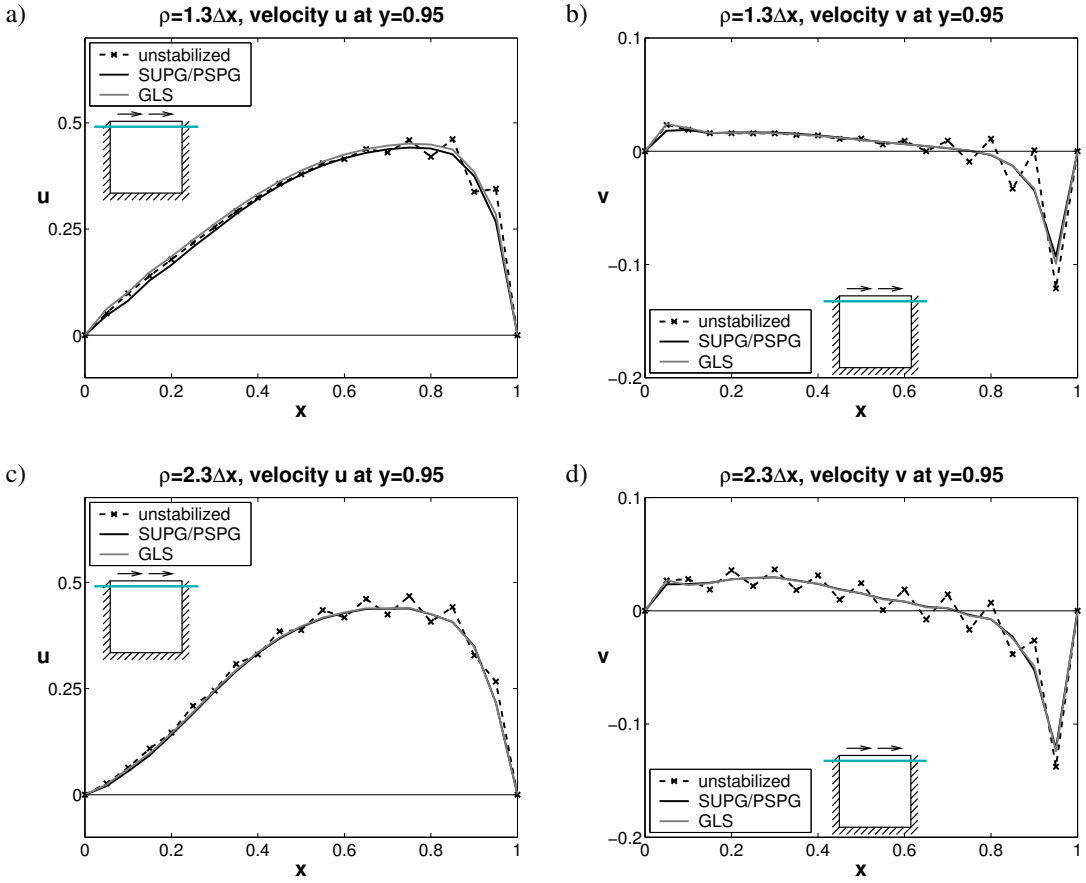

Figure 29: Velocity profiles for u and v near the tangential flow boundary at $y = 0.95$ for different dilatation parameters of $\rho = 1.3\Delta x$ and $\rho = 2.3\Delta x$.

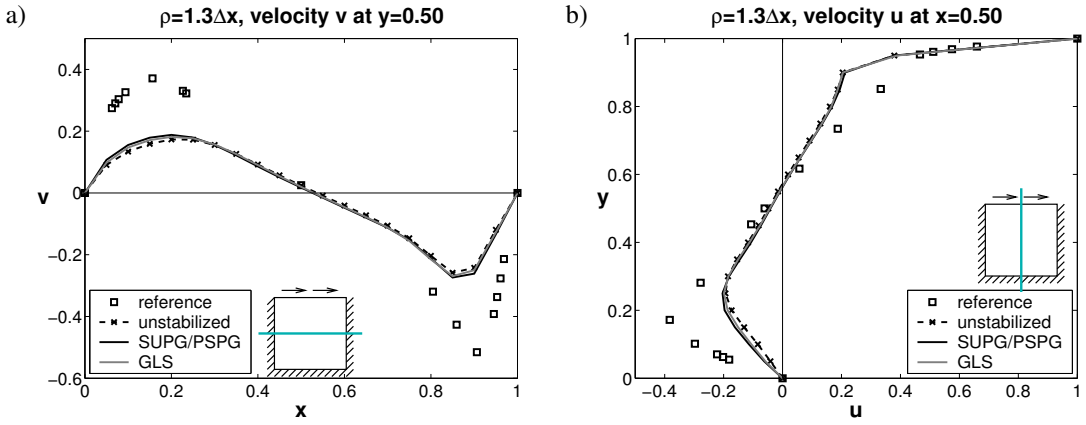

Figure 30: Velocity profiles for u and v along $y = 0.5$ and $x = 0.5$ respectively (for $\rho = 1.3\Delta x$ and 21×21 nodes).

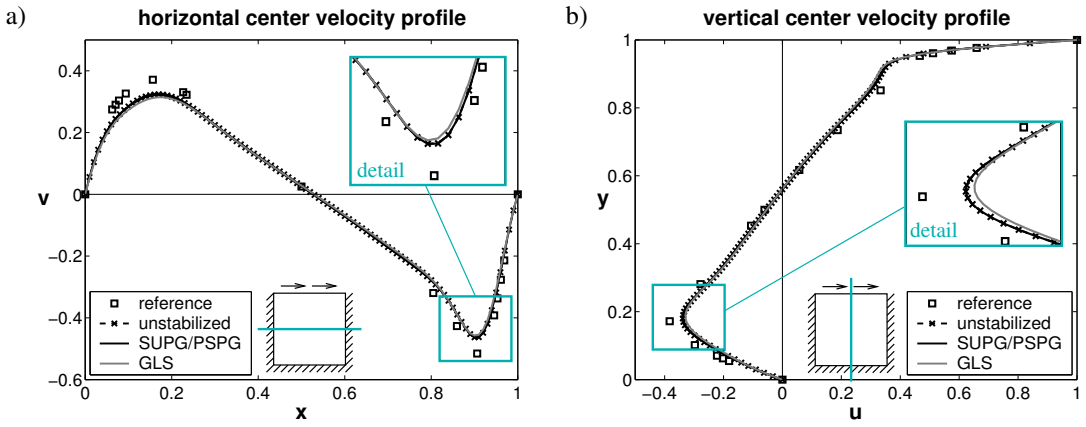

Figure 31: Velocity profiles for u and v along $y = 0.5$ and $x = 0.5$ respectively (for $\rho = 1.3\Delta x$ and 101×101 nodes). The details show that unstabilized and SUPG/PSPG-results are indistinguishable, whereas the GLS result is slightly more diffusive.

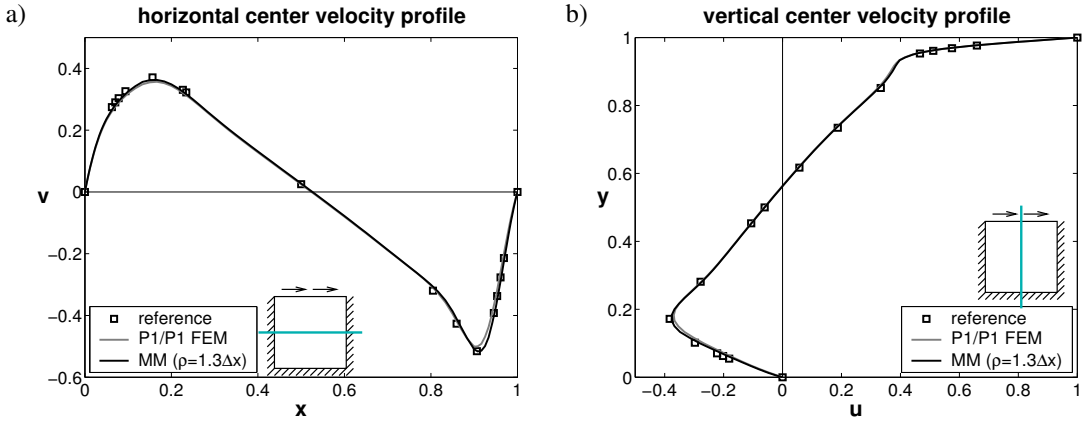

Figure 32: Velocity profiles for u and v along $y = 0.5$ and $x = 0.5$ respectively (for $\rho = 1.3\Delta x$ and 96×96 irregular nodes).

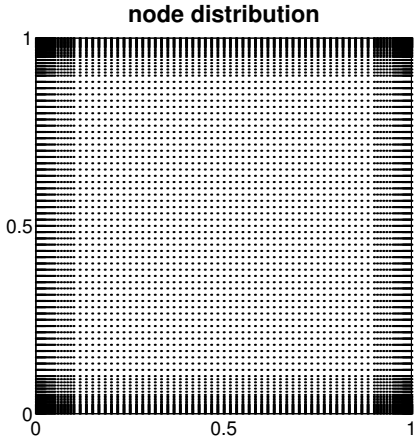

Figure 33: Node distribution with refined boundary areas for the driven cavity test case (96×96 nodes).

unknowns. The P1/P1 element is chosen for comparison because it has the same intrinsic linear basis than the MLS shape functions. For both numerical methods, SUPG/PSPG stabilization and a node distribution as shown in Fig. 33 has been used. The supports of the nodes are anisotropic with respect to the distance of the neighboring nodes,

$$\rho_{x,i} = s \cdot \min\left(|x_j - x_i|\right), \ \forall i \neq j, \tag{5.59}$$

$$\rho_{y,i} = s \cdot \min\left(|y_j - y_i|\right), \ \forall i \neq j, \tag{5.60}$$

with $s = 1.6$. The min-version (5.55) for the support length h_ρ performs best when compared to the other h_ρ-versions. A clear convergence towards the reference solution can be found, and it may be seen that the meshfree solution is more accurate than the P1/P1 element. Comparing the results for the regular 101×101 mesh with the resolved 96×96 mesh, one can clearly see the improvement in the solution for the anisotropic supports. Hence, as well as using high-aspect ratio elements in meshbased methods in order to resolve boundary layers successfully, high-aspect anisotropic supports should be used in the meshfree context analogously.

5.4.3 Cylinder Flow

The "steady-state" solution for flow past a cylinder at $\mathrm{Re} = 100$ is computed, as presented in [178]. Instationary computations at this Reynolds-number lead to periodic flow patterns

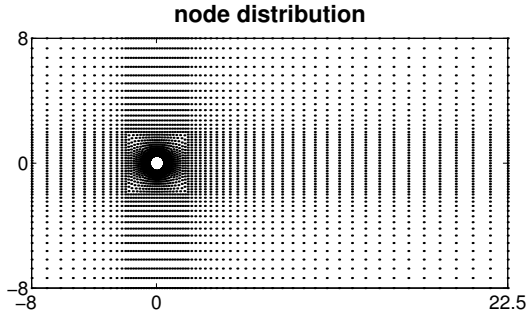

Figure 34: Irregular node distribution for the flow past a cylinder test case (6268 nodes).

known as the Kármán vortex street, which is considered in section 6. This, however, is not considered here, because at this point the focus is on the smoothing properties of the stabilization.

A channel flow with $\Omega = (-8.0, 22.5) \times (-8.0, 8.0)$ is considered, placing a cylinder with diameter 1.0 at $(0, 0)$. The situation is depicted in Fig. 34, where also the irregular node distribution for this test case is shown. Fig. 35 shows a typical result for the velocity and pressure distribution around the cylinder. The fluid parameters are $\varrho_{\mathrm{F}} = 1.0$ and $\mu = 0.001$. On the left side of the domain an inflow with $u = 0.1$ and $v = 0.0$ is prescribed, the outflow on the right boundary of the domain is realized by the traction-free boundary condition, see e.g. [70]. Along the upper and lower wall, slip boundary conditions with $v = 0.0$ are set, and no-slip boundary conditions are realized on the cylinder surface. The resulting Reynolds number of this test case is $\mathrm{Re} = 100$, when taking the cylinder diameter as a characteristic length scale.

6268 MLS nodes have been used for the computation. The supports of the meshfree shape functions are anisotropic as defined above for the irregular driven cavity test case. The computational effort for the approximation is much larger than for a comparable FEM computation, underlining the need for a coupled meshfree/meshbased method as discussed in section 6. Therefore, only stationary results are shown here, instationary computations of a cylinder flow are realized for the coupled method in section 6.5.3.

Fig. 36 depicts oscillatory unstabilized velocity profiles for u and v at $y = 5.6$. Both, SUPG/PSPG and GLS stabilization smooth out the oscillations successfully. The conclusion is that the stabilization with small dilatation parameters works successfully also for anisotropic supports.

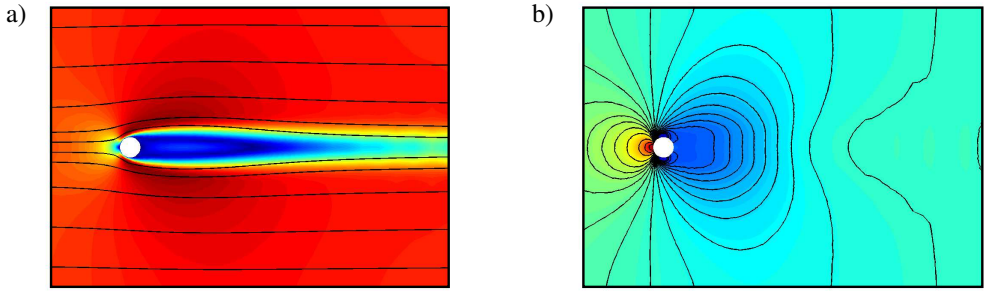

Figure 35: Detail of the stationary solution of the a) velocity and b) pressure field around the cylinder at Re = 100.

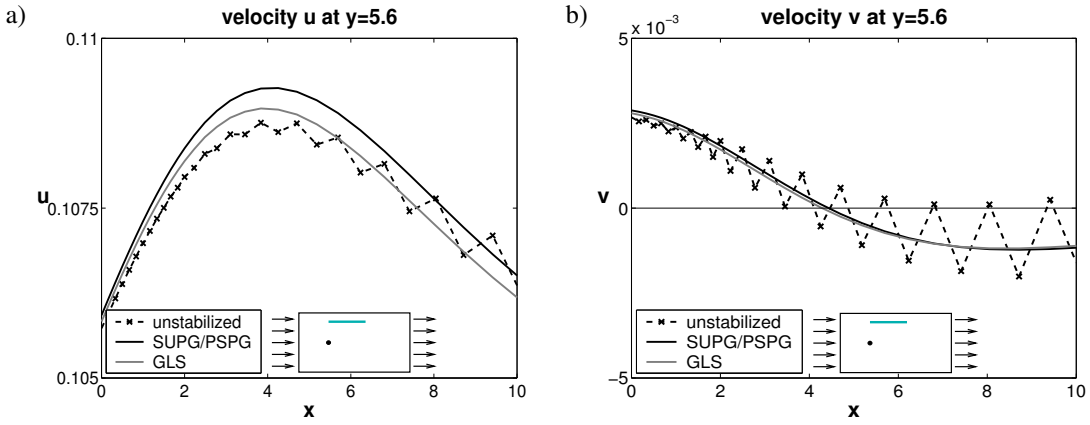

Figure 36: Velocity profiles for u and v at $y = 5.6$.

6 Coupling Meshfree and Meshbased Methods

The *coupling* of meshfree and meshbased methods is desirable in order to use meshfree methods only in small parts of the domain, where they are needed because a mesh may be particularly difficult to be maintained there, and standard meshbased methods in the rest of the domain. For this purpose several coupling approaches have been proposed e.g. in [24, 86, 138]. The coupling approach of Belytschko *et al.* [24] employs ramp functions for the blending of the meshfree and meshbased parts of the domain, very similar versions of this approach are found in [39, 131]. The approach of Huerta *et al.* [86] considers the contribution of the finite element shape functions in the computation of the meshfree shape functions. Also higher-order coupled shape functions may be obtained with these techniques. The bridging scale method of Liu *et al.* [138] may also be used to couple meshfree and meshbased shape functions. However, this approach requires meshfree and meshbased shape functions everywhere in the computational domain, thereby not reducing the computational effort of the coupled formulation. Hence, it is not considered here.

Coupling meshfree and meshbased methods has also been performed with the aim to combine other advantages of both methods. It may be desirable to introduce the favorable characteristics of meshfree methods with respect to continuity [127, 135], adaptivity [50], enrichment [104, 108] etc. Methods like the generalized finite element method [173, 174], partition of unity finite element method [145] and *hp*-clouds [48, 49, 157] may also be considered as hybrids of meshfree and meshbased methods, as they combine ideas from both areas.

For both, the meshbased and meshfree parts of the domain the weak form of the incompressible Navier-Stokes equations in Eulerian or ALE formulation [98] are approximated. This is standard for meshbased methods—where it would be almost impossible to take the Lagrangian viewpoint and maintain a conforming mesh throughout the flow simulation—and is also applied for the meshfree part in order to make the coupling as straightforward as possible. The Eulerian and ALE viewpoint require stabilization, which is discussed in detail in section 5, see also [60, 63]. There, it has been shown that standard stabilization methods may be applied to meshfree methods as well, however, with a careful choice of the stabilization parameter τ which weighs the stabilization. Only small dilatation parameters of the meshfree shape functions justify the use of standard formulas for τ.

In the context of coupled meshfree/meshbased shape functions it is found that the standard coupling approaches of [24, 86] require modifications, see also [58, 61]. The approach

of Huerta *et al.* [86] is modified in a way that smaller dilatation parameters of the meshfree shape functions are possible, being more suitable for stabilization. The approach of Belytschko *et al.* [24] is modified slightly such that the shape functions are more regular in the transition area where meshfree and meshbased functions are coupled, which is also advantageous for stabilization. The resulting stabilized and coupled formulation is validated and successfully applied to a number of test cases.

Section 6.1 starts with a review of various coupling approaches with different emphases. The approaches of [24] and [86] are described in section 6.3 and 6.4. The modifications in order to obtain coupled shape functions that are more suitable for stabilization are introduced. In section 6.5, the success of the stabilized and coupled formulation is shown starting with a convergence test of the one-dimensional advection-diffusion equation using the different coupling approaches. Then, the coupled formulations are applied to the incompressible Navier-Stokes equations. The fluid solver is validated with two standard test cases, and applied to fluid-structure interaction problems involving moving and rotating objects. All test cases show that the coupled approximations have the same order of convergence as pure FEM calculations, and that reliable and accurate solutions are obtained with the modified coupling approaches.

6.1 Coupling in Different Contexts

Coupling meshfree and meshbased methods has been realized in many different ways. The aim is always to combine certain advantages of each method. The following examples are found:

Continuity: Meshfree shape functions may be constructed to have any desired order of continuity, see e.g. [56]. In contrast, meshbased shape functions are often only C^0-continuous in the domain. The construction of higher-order continuous finite element shape functions in multi-dimensions poses serious problems, see e.g. [127]. With the aim to construct element shape functions with any desired order of continuity, Li, Liu *et al.* introduce the reproducing kernel *element* method, see [135] and [127].

Adaptivity: The absence of a mesh in meshfree methods is advantageous for adaptive strategies. Only nodes have to be added or removed where desired, without the need to keep a conforming mesh. In [50], Fernandéz *et al.* make use of this fact and introduce meshfree areas in a FEM domain where adaptivity is desired.

Enrichment: The enrichment of the approximation space with certain functions may drastically improve the convergence properties of a numerical method. This is comparably easy possible with some meshfree methods such as the generalized finite element method (GFEM) [173, 174], partition of unity finite element method (PUFEM) [145] and *hp*-clouds [48, 49, 157]. These methods combine ideas from the FEM and MMs. More direct approaches for coupling meshfree and meshbased methods for enrichment may be found in [86, 138].

Meshing: In problems involving large geometric deformations, moving boundaries, or moving and rotating objects, maintaining of a conforming mesh is often very difficult. Furthermore, the costs for frequent remeshing—which may even fail in complex geometric situations—are not negligible, and projection errors between the meshes are introduced [104]. Thus, it may be desirable to employ meshfree shape functions in parts of the domain, where a mesh causes problems, and meshbased shape functions in the remaining area. Coupling approaches for meshbased and meshfree shape functions may be found in Belytschko *et al.* [24], in Huerta *et al.* [86], and in Liu *et al.* [138, 185]. Other ways are shown in [104, 108]. The meshing aspect is closely related to connectivity: Sometimes the connectivity of the nodes in parts of the domain changes during runtime (e.g. in case of a rotating object), then it may be desirable to use meshfree shape functions there, because they compute the connectivity at run-time, in contrast to meshbased methods which define the connectivity *a priori* with a mesh.

Computational effort: Meshfree shape functions are comparably expensive to compute. The functions are of a highly non-polynomial character, which makes integration in a Galerkin setting demanding. Large numbers of integration points are necessary, and at each integration point a small system of equations ($\mathbf{M}(\boldsymbol{x})$) has to be built up—including a neighbour search—and inverted in order to determine the meshfree shape functions. The computation of the shape functions' derivatives involves matrix-vector operations whose costs are not negligible. Therefore, it is often desirable from a computational viewpoint to use meshfree shape functions as little as possible. Consequently, the aim is to employ meshfree shape functions where their properties are desirable—according to any of the previous aspects—and meshbased shape function in the rest of the domain.

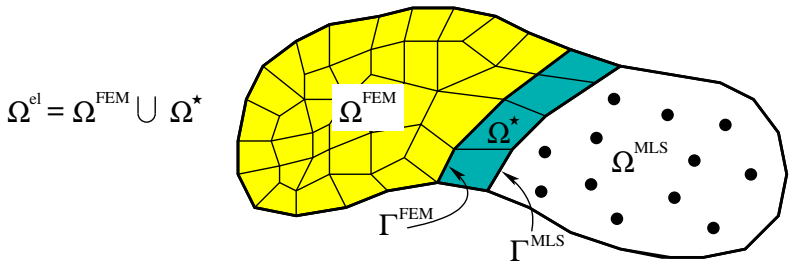

Figure 37: Decomposition of the domain into Ω^{FEM}, Ω^{MLS} and $\Omega^{\star}$.

Here, the aim is to develop a coupled meshfree/meshbased fluid solver which is able to simulate complex flow phenomena including large geometric deformations and moving and rotating obstacles. Therefore, the meshing aspect together with the consideration of the computational effort is important. Thus, the approaches of Belytschko *et al.* [24] and Huerta *et al.* [86] for coupling meshbased and meshfree shape functions may be chosen. The approach in [24] employs ramp functions in the transition area between the purely meshfree and meshbased parts of the domain, whereas the approach in [86] modifies the consistency conditions of the MLS procedure considering the contributions of the finite element shape functions in the transition area. The approach of [138, 185] (bridging scale method) is not considered, as there—due to continuity requirements—the coupling may only be performed for meshfree and meshbased shape functions defined everywhere in the domain, not leading to a reduced computational cost.

6.2 Preliminaries

For a coupling of meshfree and meshbased shape functions, the domain Ω is decomposed into disjoint domains Ω^{el} and Ω^{MLS}, with the common boundary Γ^{MLS}. The domain Ω^{el} is discretized with standard quadrilateral finite elements. The union of all elements along Γ^{MLS} is called the transition area $\Omega^{\star}$, so that Ω^{el} may further be decomposed into the disjoint domains Ω^{FEM} and $\Omega^{\star}$, connected by a boundary labeled Γ^{FEM}; clearly $\Omega^{\mathrm{FEM}} \bigcap \Omega^{\mathrm{MLS}} = \emptyset$. This situation is sketched in Fig. 37.

Throughout this work, consistency of first order is fulfilled by the set of meshbased, meshfree and coupled shape functions. This results in the ability of reproducing linear solutions *exactly*.

6.3 Coupling with Consistency Conditions

6.3.1 Original Approach

The coupling approach of Huerta *et al.* [86] considers the contributions of the meshbased FEM shape functions in the computation of the MLS shape functions by modified consistency conditions; see [56] for an alternative deduction of this coupling approach. The resulting coupled set of shape functions is consistent up to the desired order.

In the original approach [86], FEM nodes are placed in the standard way in the elements inside Ω^{FEM}, however *not* in $\Omega^{\star}\backslash\Gamma^{\mathrm{FEM}}$. The corresponding meshbased shape functions of the FEM nodes *remain unchanged*, and the coupling is considered only in the meshfree shape functions. Meshfree nodes with corresponding supports $\widetilde{\Omega}_i$ may be arbitrarily distributed in Ω^{MLS} and $\Omega^{\star}$. It is helpful to introduce the complementary nodal sets

$$
\begin{aligned}
I^{\mathrm{FEM}} &= \left\{ i \,\middle|\, \boldsymbol{x}_i \in \Omega^{\mathrm{FEM}} \setminus \Gamma^{\mathrm{FEM}} \right\}, && (6.1)\\
I^{\mathrm{MLS}} &= \left\{ i \,\middle|\, \widetilde{\Omega}_i \subset \Omega^{\mathrm{MLS}} \right\}, && (6.2)\\
I^{\star} &= \left\{ i \,\middle|\, \widetilde{\Omega}_i \bigcap \Omega^{\mathrm{el}} \neq \emptyset,\ \boldsymbol{x}_i \notin \Gamma^{\mathrm{FEM}} \right\}, && (6.3)\\
I^{\star\star} &= \left\{ i \,\middle|\, \boldsymbol{x}_i \in \Gamma^{\mathrm{FEM}} \right\}. && (6.4)
\end{aligned}
$$

In words, I^{MLS} is the set of meshfree nodes whose supports are fully inside Ω^{MLS}, and $I^{\star}$ is the set of MLS nodes that have supports overlapping with elements. Then, the meshbased and meshfree shape functions for the nodes at $\boldsymbol{x}_i$ are computed as [86]

$$
\begin{aligned}
\forall i \in I^{\mathrm{FEM}} \cup I^{\star\star} :\ N_i &= N_i^{\mathrm{FEM}}, && (6.5)\\
\forall i \in I^{\mathrm{MLS}} :\ N_i &= N_i^{\mathrm{MLS}} = \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) \left[\mathbf{M}(\boldsymbol{x})\right]^{-1} \phi(\boldsymbol{x}-\boldsymbol{x}_i)\, \boldsymbol{p}(\boldsymbol{x}_i), && (6.6)\\
\forall i \in I^{\star} :\ N_i &= \left(\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) - \sum_{j \in I^{\mathrm{FEM}}} N_j^{\mathrm{FEM}}(\boldsymbol{x})\, \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}_j) \right) \cdot && (6.7)\\
&\qquad \left[\mathbf{M}(\boldsymbol{x})\right]^{-1} \phi(\boldsymbol{x}-\boldsymbol{x}_i)\, \boldsymbol{p}(\boldsymbol{x}_i).
\end{aligned}
$$

N_i^{FEM} are the standard bi-linear finite element shape functions, and N_i^{MLS} are the standard MLS functions, defined in section 4.3. The moment matrix $\mathbf{M}(\boldsymbol{x})$ from (4.22) is adjusted

to the coupled situation

$$\mathbf{M}(\boldsymbol{x}) = \sum_{i \in I'} \phi(\boldsymbol{x} - \boldsymbol{x}_i)\, \boldsymbol{p}(\boldsymbol{x}_i)\, \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}_i), \quad I' = I^{\mathrm{MLS}} \cup I^{\star} \cup I^{\star\star}. \tag{6.8}$$

Fig. 38 shows the sets I^{FEM}, I^{MLS}, $I^{\star}$ and $I^{\star\star}$ and displays the resulting shape functions of this approach in a section of a one-dimensional domain with a regular node distribution around the transition area $\Omega^{\star}$.

6.3.2 Modification

Instead of keeping the FEM shape function unchanged inside the transition area as in the original approach, one may additionally place meshfree nodes at the FEM node positions along Γ^{FEM} and superimpose the two shape functions at these nodes. The modified shape functions result as follows

$$\forall i \in I^{\mathrm{FEM}}: \; N_i = N_i^{\mathrm{FEM}}, \tag{6.9}$$

$$\forall i \in I^{\mathrm{MLS}}: \; N_i = N_i^{\mathrm{MLS}}, \tag{6.10}$$

$$\forall i \in I^{\star}: \; N_i = N_i^{\star} = \Big(\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) - \sum_{j \in I^{\mathrm{FEM}} \cup I^{\star\star}} N_j^{\mathrm{FEM}}(\boldsymbol{x})\, \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}_j)\Big) \cdot \left[\mathbf{M}(\boldsymbol{x})\right]^{-1} w(\boldsymbol{x} - \boldsymbol{x}_i)\, \boldsymbol{p}(\boldsymbol{x}_i), \tag{6.11}$$

$$\forall i \in I^{\star\star}: \; N_i = N_i^{\star\star} = \Big(\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) - \sum_{j \in I^{\mathrm{FEM}} \cup I^{\star\star}} N_j^{\mathrm{FEM}}(\boldsymbol{x})\, \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}_j)\Big) \cdot \left[\mathbf{M}(\boldsymbol{x})\right]^{-1} w(\boldsymbol{x} - \boldsymbol{x}_i)\, \boldsymbol{p}(\boldsymbol{x}_i) + N_i^{\mathrm{FEM}}. \tag{6.12}$$

The right part of Fig. 38 shows the resulting shape functions of the modified approach of [86]. The important advantage of this modification is that smaller dilatation parameters are possible (although in this figure $\rho_i = 2.9\Delta x$ has been taken for both approaches). For example, in the original approach and a regular distribution of nodes in one dimension, one finds that in case of *linear* consistency for the regularity of the matrix $\mathbf{M}(\boldsymbol{x})$, dilatation parameters of $\rho_i > 2.0\Delta x$ are required [86]. This follows from the requirement for the regularity of $\mathbf{M}(\boldsymbol{x})$ [86, 137], see section 4.3.4, that

$$\mathrm{card}\left\{\boldsymbol{x}_i \middle| \, w(\boldsymbol{x} - \boldsymbol{x}_i) \neq 0 \; \forall i \in I''\right\} \geq k = \dim(\mathbf{M}), \tag{6.13}$$

with $I'' = I^{\mathrm{MLS}} \cup I^{\star}$, which near the boundary Γ^{FEM} can only be fulfilled with $\rho_i > 2.0\Delta x$.

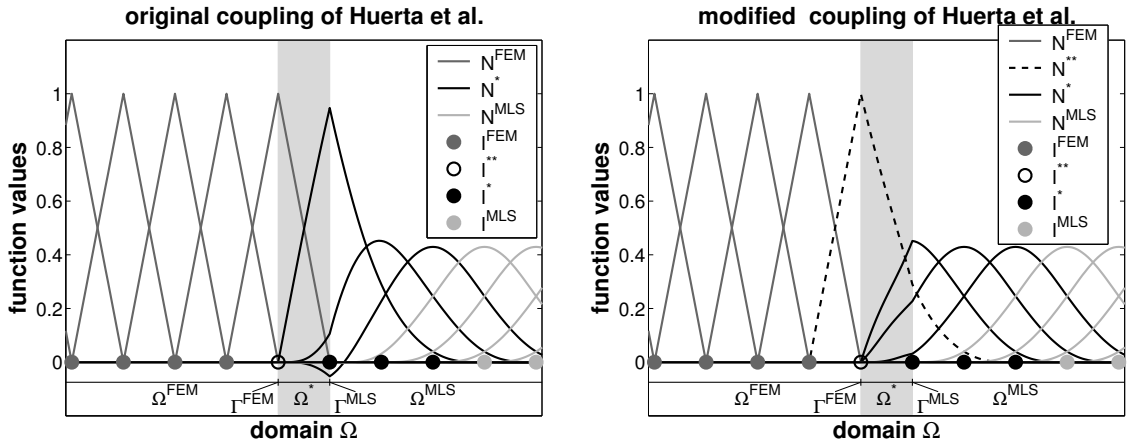

Figure 38: Shape functions of the coupling approach of Huerta in the original [86] and modified version.

In contrast, with the modified approach the nodes along Γ^{FEM} are added to I'', hence $I'' = I^{\mathrm{MLS}} \cup I^{\star} \cup I^{\star}$, and thus $\rho_i > 1.0\Delta x$ are sufficient for linear consistency. This holds analogously in multidimensional domains and is an important advantage for the reliable stabilization of (non-linear) partial differential equations, see section 5.

6.3.3 Proof of Consistency

In this section the consistency of the shape functions in the domain is proven. Consistency is often shown by help of a Taylor series expansion. Applying the Taylor series (4.28) for a general approximation (3.2) gives

$$u(\boldsymbol{x}) \approx u^h(\boldsymbol{x}) = \sum_{i=1}^{r} N_i(\boldsymbol{x})\, u(\boldsymbol{x}_i)\,, \tag{6.14}$$

$$= \sum_{i=1}^{r} N_i(\boldsymbol{x}) \sum_{|\boldsymbol{\alpha}|=0}^{\infty} \frac{(\boldsymbol{x}_i - \boldsymbol{x})^{\boldsymbol{\alpha}}}{|\boldsymbol{\alpha}|!} D^{\boldsymbol{\alpha}} u(\boldsymbol{x})\,. \tag{6.15}$$

The function $u(\boldsymbol{x})$ is on the left and right hand side ($D^{\boldsymbol{\alpha}^1} u(\boldsymbol{x}) = D^{(0,\ldots,0)} u(\boldsymbol{x}) = u(\boldsymbol{x})$), but derivatives of $u(\boldsymbol{x})$ occur only on the right hand side. Thus, comparing the coefficients

leads to the infinite system of equations

$$\sum_{i=1}^{r} N_i(\boldsymbol{x}) \frac{(\boldsymbol{x}_i - \boldsymbol{x})^{\boldsymbol{\alpha}^1}}{|\boldsymbol{\alpha}^1|!} = 1, \tag{6.16}$$

$$\sum_{i=1}^{r} N_i(\boldsymbol{x}) \frac{(\boldsymbol{x}_i - \boldsymbol{x})^{\boldsymbol{\alpha}^s}}{|\boldsymbol{\alpha}^s|!} = 0, \quad s = \{2, 3, \ldots\} \tag{6.17}$$

which, using (4.4) with $n = \infty$, may be shortly written as

$$\sum_{i=1}^{r} N_i(\boldsymbol{x}) \boldsymbol{p}(\boldsymbol{x}_i - \boldsymbol{x}) = \boldsymbol{p}(0). \tag{6.18}$$

Depending on the number of equations that can be fulfilled by the shape functions N_i, a certain order of consistency may be proven. One may see that (6.18) is simply the standard consistency condition (4.1) with a shifted argument $-\boldsymbol{x}$ of the basis. After showing this analogy of the standard consistency condition with the Taylor series approach, the proof is further carried out using the standard conditions (4.1). It is first shown that (4.1) is fulfilled for the coupled shape function situation in $\Omega^\star$.

$$\sum_{i=1}^{r} N_i(\boldsymbol{x}) \boldsymbol{p}(\boldsymbol{x}_i) = \boldsymbol{p}(\boldsymbol{x}), \tag{6.19}$$

$$\sum_{i \in I^{\star\star}} N_i^{\star\star} \boldsymbol{p}(\boldsymbol{x}_i) + \sum_{i \in I^{\star}} N_i^{\star} \boldsymbol{p}(\boldsymbol{x}_i) + \sum_{i \in I^{\mathrm{FEM}}} N_i^{\mathrm{FEM}} \boldsymbol{p}(\boldsymbol{x}_i) = \boldsymbol{p}(\boldsymbol{x}). \tag{6.20}$$

N_i^{MLS} is not considered because it is 0 everywhere in $\Omega^\star$. Applying the definitions of the shape functions in (6.9)–(6.12) gives

$$\begin{aligned} \sum_{i \in I^{\star\star}} \Big(\boldsymbol{z}^{\mathrm{T}}(\boldsymbol{x}) \left[\mathbf{M}(\boldsymbol{x})\right]^{-1} w(\boldsymbol{x} - \boldsymbol{x}_i) \boldsymbol{p}(\boldsymbol{x}_i) + N_i^{\mathrm{FEM}} \Big) \boldsymbol{p}(\boldsymbol{x}_i) + \\ \sum_{i \in I^{\star}} \Big(\boldsymbol{z}^{\mathrm{T}}(\boldsymbol{x}) \left[\mathbf{M}(\boldsymbol{x})\right]^{-1} w(\boldsymbol{x} - \boldsymbol{x}_i) \boldsymbol{p}(\boldsymbol{x}_i) \Big) \boldsymbol{p}(\boldsymbol{x}_i) + \\ \sum_{i \in I^{\mathrm{FEM}}} N_i^{\mathrm{FEM}} \boldsymbol{p}(\boldsymbol{x}_i) = \boldsymbol{p}(\boldsymbol{x}), \end{aligned} \tag{6.21}$$

where

$$\boldsymbol{z}(\boldsymbol{x}) = \boldsymbol{p}(\boldsymbol{x}) - \sum_{j \in I^{\mathrm{FEM}} \cup I^{\star\star}} N_j^{\mathrm{FEM}}(\boldsymbol{x}) \boldsymbol{p}(\boldsymbol{x}_j). \tag{6.22}$$

This is rearranged using the symmetry of $\mathbf{M}(\boldsymbol{x})$, hence $\boldsymbol{z}^{\mathrm{T}}\mathbf{M}\boldsymbol{p}\boldsymbol{p} = \boldsymbol{p}\boldsymbol{p}^{\mathrm{T}}\mathbf{M}\boldsymbol{z}$,

$$\begin{aligned}\sum_{i\in I^{\star\star}}\Big(\phi(\boldsymbol{x}-\boldsymbol{x}_i)\,\boldsymbol{p}(\boldsymbol{x}_i)\,\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}_i)\Big)\,[\mathbf{M}(\boldsymbol{x})]^{-1}\,\boldsymbol{z}(\boldsymbol{x}) + \sum_{i\in I^{\star\star}} N_i^{\mathrm{FEM}}\boldsymbol{p}(\boldsymbol{x}_i) + \\ \sum_{i\in I^{\star}}\Big(\phi(\boldsymbol{x}-\boldsymbol{x}_i)\,\boldsymbol{p}(\boldsymbol{x}_i)\,\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}_i)\Big)\,[\mathbf{M}(\boldsymbol{x})]^{-1}\,\boldsymbol{z}(\boldsymbol{x}) + \\ \sum_{i\in I^{\mathrm{FEM}}} N_i^{\mathrm{FEM}}\boldsymbol{p}(\boldsymbol{x}_i) \;=\; \boldsymbol{p}(\boldsymbol{x}),\end{aligned} \tag{6.23}$$

which further reduces to

$$\begin{aligned}\sum_{i\in I^{\star}\cup I^{\star\star}}\Big(\phi(\boldsymbol{x}-\boldsymbol{x}_i)\,\boldsymbol{p}(\boldsymbol{x}_i)\,\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}_i)\Big)\,[\mathbf{M}(\boldsymbol{x})]^{-1}\,\boldsymbol{z}(\boldsymbol{x}) + \\ \sum_{i\in I^{\mathrm{FEM}}\cup I^{\star\star}} N_i^{\mathrm{FEM}}\boldsymbol{p}(\boldsymbol{x}_i) \;=\; \boldsymbol{p}(\boldsymbol{x}).\end{aligned} \tag{6.24}$$

The left expression is directly the definition of the moment matrix (6.8) ($I^{\mathrm{MLS}} = \emptyset$ for the transition area), consequently $\mathbf{M}(\boldsymbol{x})\cdot[\mathbf{M}(\boldsymbol{x})]^{-1} = \mathbf{I}$, so that by using (6.22) the proof of consistency in $\Omega^{\star}$ may be completed as

$$\mathbf{I}\boldsymbol{z}(\boldsymbol{x}) + \sum_{i\in I^{\mathrm{FEM}}\cup I^{\star\star}} N_i^{\mathrm{FEM}}\boldsymbol{p}(\boldsymbol{x}_i) \;=\; \boldsymbol{p}(\boldsymbol{x}), \tag{6.25}$$

$$\boldsymbol{p}(\boldsymbol{x}) - \sum_{j\in I^{\mathrm{FEM}}\cup I^{\star\star}} N_j^{\mathrm{FEM}}(\boldsymbol{x})\,\boldsymbol{p}(\boldsymbol{x}_j) + \sum_{i\in I^{\mathrm{FEM}}\cup I^{\star\star}} N_i^{\mathrm{FEM}}\boldsymbol{p}(\boldsymbol{x}_i) \;=\; \boldsymbol{p}(\boldsymbol{x}), \tag{6.26}$$

$$\boldsymbol{p}(\boldsymbol{x}) \;=\; \boldsymbol{p}(\boldsymbol{x}). \tag{6.27}$$

The consistency in the region Ω^{MLS} may be shown easily by noting that no influence of the finite element shape functions is apparent there. That is, the shape functions are directly the original MLS shape functions of section 4.3.1 and the consistency condition (4.1) for this situation may be stated as

$$\sum_{i=1}^{r} N_i(\boldsymbol{x})\,\boldsymbol{p}(\boldsymbol{x}_i) = \sum_{i\in I'} N_i^{\mathrm{MLS}}(\boldsymbol{x})\,\boldsymbol{p}(\boldsymbol{x}_i) \;=\; \boldsymbol{p}(\boldsymbol{x}), \tag{6.28}$$

$$\sum_{i\in I'} \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x})\,[\mathbf{M}(\boldsymbol{x})]^{-1}\,\phi(\boldsymbol{x}-\boldsymbol{x}_i)\,\boldsymbol{p}(\boldsymbol{x}_i)\,\boldsymbol{p}(\boldsymbol{x}_i) \;=\; \boldsymbol{p}(\boldsymbol{x}), \tag{6.29}$$

with $I' = I^{\mathrm{MLS}} \cup I^{\star} \cup I^{\star\star}$. Rearranging (6.29) as shown above and using (6.8) directly gives

$$\mathbf{M}(\boldsymbol{x}) \left[\mathbf{M}(\boldsymbol{x})\right]^{-1} \boldsymbol{p}(\boldsymbol{x}) \;=\; \boldsymbol{p}(\boldsymbol{x}) \,. \tag{6.30}$$

Finally, the consistency in Ω^{FEM} is discussed, where only standard finite element shape functions are present. It is pointed out that the bi-linear shape functions (as well as other finite element shape functions) may also be obtained by a least-squares principle. That is, one may write these functions in the same form of

$$N_i^{\mathrm{FEM}}(\boldsymbol{x}) = \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) \left[\mathbf{M}^{\mathrm{FEM}}\right]^{-1} \phi^{\mathrm{FEM}} \boldsymbol{p}(\boldsymbol{x}_i) \tag{6.31}$$

than the MLS shape functions, compare with (4.25). However, the definition of the weighting function is different, which also leads to a different moment matrix $\mathbf{M}^{\mathrm{FEM}}(\boldsymbol{x})$. For bi-linear shape functions, the weighting function of a node is $\phi^{\mathrm{FEM}} = 1$ in the union of the quadrilateral elements around that node. The element shape functions in an arbitrary element with corners in $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{x}_3$, $\boldsymbol{x}_4$ may be obtained by (6.31) with

$$\mathbf{M}^{\mathrm{FEM}} \;=\; \sum_{i=1}^{4} 1 \cdot \boldsymbol{p}(\boldsymbol{x}_i)\, \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}_i) \,, \tag{6.32}$$

$$\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) \;=\; (1, x, y, xy) \,. \tag{6.33}$$

Consequently, the proof of consistency of (6.31) follows directly the principle of the proof for the MLS shape functions.

6.4 Coupling by a Ramp Function

6.4.1 Original Approach

In the approach of Belytschko *et al.* [24], meshfree and meshbased shape functions for the nodes are computed and defined independently and coupled with help of a ramp function $R(\boldsymbol{x})$. FEM nodes are placed in Ω^{el}, in contrast to the coupling approach of [86], where they are placed in Ω^{FEM} only. Meshfree nodes with supports $\widetilde{\Omega}_i$ are distributed arbitrarily in $\Omega^{\star} \bigcup \Omega^{\mathrm{MLS}}$ and may also be included inside Ω^{FEM}. The latter will only affect the shape functions inside $\Omega^{\star} \bigcup \Omega^{\mathrm{MLS}}$, i.e. MLS nodes with $\widetilde{\Omega}_i \subset \Omega^{\mathrm{FEM}}$ have no influence at all.

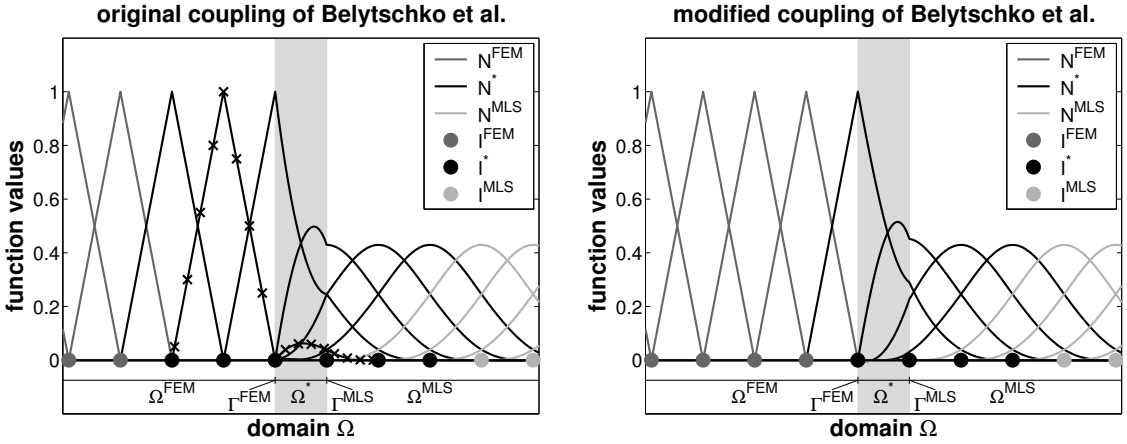

Figure 39: Shape functions of the coupling approach of Belytschko in the original [24] and modified version.

Belytschko *et al.* [24] define the ramp function as follows:

$$R(\boldsymbol{x}) = \begin{cases} 0 & , \boldsymbol{x} \in \Omega^{\mathrm{FEM}} \\ 1 & , \boldsymbol{x} \in \Omega^{\mathrm{MLS}} \\ \sum_{i \in I'} N_i^{\mathrm{FEM}}(\boldsymbol{x}) & , \boldsymbol{x} \in \Omega^{\star}, I' = \left\{ i \,\middle|\, \boldsymbol{x}_i \in \Gamma^{\mathrm{MLS}} \right\}, \end{cases} \tag{6.34}$$

i.e. it varies monotonically between 0 and 1 in $\Omega^{\star}$. The consistency of the resulting set of meshbased, meshfree and coupled shape functions is maintained [24].

The shape functions are defined as follows:

$$\forall i \in I^{\mathrm{FEM}} : \; N_i = N_i^{\mathrm{FEM}}, \tag{6.35}$$

$$\forall i \in I^{\mathrm{MLS}} : \; N_i = N_i^{\mathrm{MLS}}, \tag{6.36}$$

$$\forall i \in I^{\star} : \; N_i = [1 - R(\boldsymbol{x})] N_i^{\mathrm{FEM}} + R(\boldsymbol{x}) N_i^{\mathrm{MLS}}, \tag{6.37}$$

with the nodal sets

$$I^{\mathrm{FEM}} = \left\{ i \,\middle|\, \boldsymbol{x}_i \in \Omega^{\mathrm{FEM}}, \widetilde{\Omega}_i \subset \Omega^{\mathrm{FEM}} \right\}, \tag{6.38}$$

$$I^{\mathrm{MLS}} = \left\{ i \,\middle|\, \widetilde{\Omega}_i \subset \Omega^{\mathrm{MLS}} \right\}, \tag{6.39}$$

$$I^{\star} = \left\{ i \,\middle|\, \widetilde{\Omega}_i \bigcap \Omega^{\mathrm{el}} \neq \emptyset \right\}. \tag{6.40}$$

The resulting shape functions are shown in the left part of Fig. 39. One may note that

meshfree nodes inside Ω^{FEM} have an undesirable influence in $\Omega^{\star} \bigcup \Omega^{\mathrm{MLS}}$, leading to a numerically awkward form of the shape function (cross-marked line in Fig. 39). Especially in the context of stabilization, it is neither clear from a mathematical viewpoint nor from an empirical viewpoint how to choose suitable stabilization parameters for these shape functions, see [60].

6.4.2 Modification

In order to avoid these numerically awkward shape functions, this approach is modified slightly. The meshfree nodes are restricted to the area $\Omega^{\star} \bigcup \Omega^{\mathrm{MLS}}$. The resulting shape functions are then (for sufficiently small dilatation parameters) suitable to motivate the use of standard stabilization parameters [60]. Consequently, a reliable stabilization is obtained. The resulting shape functions of the modified coupling approach of [24] are shown in the right part of Fig. 39.

Comparison of the nodal sets. Fig. 40 compares the nodal sets in the different versions of the coupling approaches in a two-dimensional domain. It may be seen that the difference between the original approaches in [86] and [24] are that FEM nodes in the former publication are distributed in Ω^{FEM} only, whereas in the latter FEM nodes are in $\Omega^{\mathrm{el}} = \Omega^{\mathrm{FEM}} \bigcup \Omega^{\star}$. Furthermore, MLS nodes are restricted to $\Omega^{\mathrm{MLS}} \bigcup \Omega^{\star}$ in the approach in [86], but may be distributed everywhere in Ω in the approach of [24] (however, an influence will only be in $\Omega^{\star} \bigcup \Omega^{\mathrm{MLS}}$). The difference between the modified versions and the original versions are as follows: In the case of the coupling of [86], MLS nodes are also distributed at the FEM nodes along Γ^{FEM} and are superimposed according to (6.11). In the case of the coupling in [24], the MLS nodes are restricted to $\Omega^{\star} \bigcup \Omega^{\mathrm{MLS}}$ only, because MLS nodes in Ω^{FEM} may show an undesirable influence in $\Omega^{\star} \bigcup \Omega^{\mathrm{MLS}}$.

Continuity of the Coupled Shape Functions. In Figs. 38 and 39 it can be seen that the coupled shape functions are only C^0-continuous along the boundary Γ^{MLS}. It is easily possible to construct shape functions that are C^1-continuous there and, consequently, restrict C^0-continuous shape functions to the area Ω^{FEM}. This can be achieved in the approach in [86] by replacing the bi-linear element shape functions in $\Omega^{\star}$ by standard Hermitian C^1-continuous element shape functions [191]. The consistency of the resulting set of coupled shape functions is still ensured due to the modified consistency conditions of

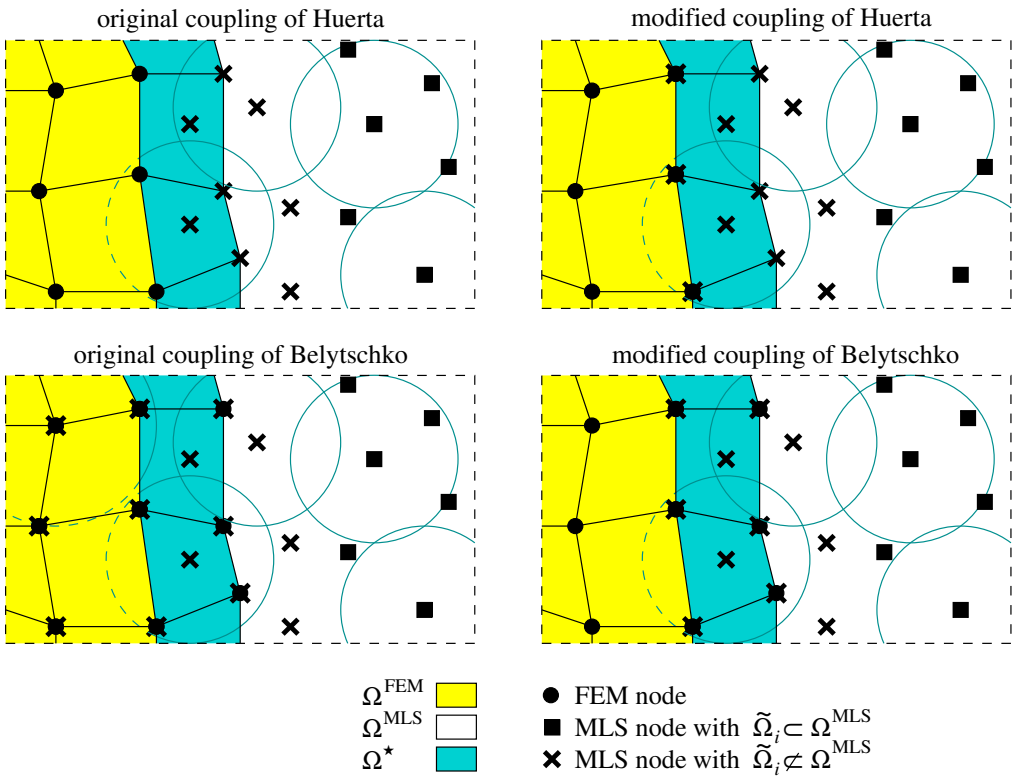

Figure 40: Differences between the nodal sets in the original and modified coupling versions. The grey circles show the supports $\widetilde{\Omega}_i$ of some MLS nodes.

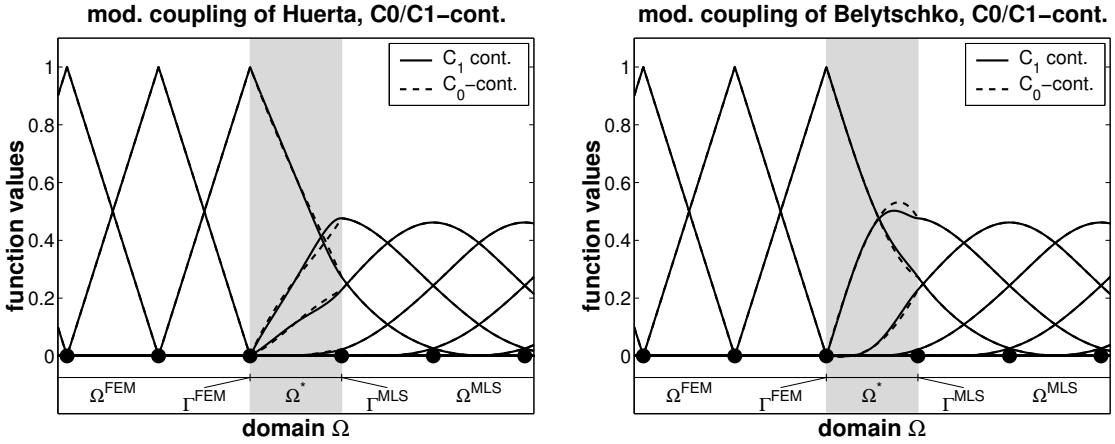

Figure 41: Comparison of C^0 and C^1 continuous shape functions along Γ^{MLS} in the modified coupling approaches.

this approach. In the coupling approach of [24], C^1-continuity along Γ^{FEM} can be achieved by using C^1-continuous ramp functions in $\Omega^\star$. Again, Hermitian element shape functions may be used. Fig. 41 shows the resulting C^1-continuous shape functions and compares them with the C^0-continuous case. Clearly, only the shape functions in $\Omega^\star$ are effected by the modifications. It is interesting to note that in own numerical studies the higher order of continuity does not lead to superior results, therefore, throughout this paper only C^0-continuity along Γ^{FEM} is fulfilled by the shape functions.

6.4.3 Proof of Consistency

In the original work [24], Belytschko *et al.* use the coupling approach with first order consistency, see (4.1), for the coupling with bi-linear finite element shape functions. The ramp function is also defined by help of bi-linear shape functions in the transition area. The generalization to higher orders is straightforward, so the proof of consistency is shown for the general case. In the modified version, meshfree nodes build a partition of unity in $\Omega^\star \bigcup \Omega^{\mathrm{MLS}}$ and FEM nodes in $\Omega^\star \bigcup \Omega^{\mathrm{FEM}}$. Consequently, the overlap of the PUs is in the transition area $\Omega^\star$, where the ramp function varies in-between 0 and 1. It is now shown

that

$$\sum_{i=1}^{r} N_i(\boldsymbol{x})\,\boldsymbol{p}(\boldsymbol{x}_i) = \boldsymbol{p}(\boldsymbol{x}), \tag{6.41}$$

$$\sum_{i=1}^{r} \Big([1 - R(\boldsymbol{x})]\, N_i^{\mathrm{FEM}} + R(\boldsymbol{x})\, N_i^{\mathrm{MLS}} \Big) \boldsymbol{p}(\boldsymbol{x}_i) = \boldsymbol{p}(\boldsymbol{x}), \tag{6.42}$$

where $N_i^{\mathrm{FEM}} = 0$ for all $\boldsymbol{x}_i \in \Omega^{\mathrm{MLS}} \setminus \Gamma^{\mathrm{MLS}}$ and $N_i^{\mathrm{MLS}} = 0$ for all $\boldsymbol{x}_i \in \Omega^{\mathrm{FEM}} \setminus \Gamma^{\mathrm{FEM}}$. A few modifications show the desired result

$$[1 - R(\boldsymbol{x})] \sum_{i=1}^{r} N_i^{\mathrm{FEM}} \boldsymbol{p}(\boldsymbol{x}_i) + R(\boldsymbol{x}) \sum_{i=1}^{r} N_i^{\mathrm{MLS}} \boldsymbol{p}(\boldsymbol{x}_i) = \boldsymbol{p}(\boldsymbol{x}), \tag{6.43}$$

$$[1 - R(\boldsymbol{x})]\, \boldsymbol{p}(\boldsymbol{x}) + R(\boldsymbol{x})\, \boldsymbol{p}(\boldsymbol{x}) = \boldsymbol{p}(\boldsymbol{x}), \tag{6.44}$$

$$\boldsymbol{p}(\boldsymbol{x}) = \boldsymbol{p}(\boldsymbol{x}), \tag{6.45}$$

where (6.44) follows from the fact that the FEM and MLS shape functions build PUs, i.e. they fulfill the consistency conditions, see also section 6.3.3.

6.5 Numerical results

6.5.1 One-dimensional Advection-Diffusion Equation

The aim of this subsection is to show that the same order of convergence is obtained with all coupling approaches. The obtained convergence of order 2 in the L_2-norm is equivalent to purely meshbased computations with linear finite elements only. The test case is defined as follows.

The one-dimensional advection-diffusion equation (5.28) in SUPG-stabilized weak form [31] is approximated in $\Omega = (0, 1)$, with a right hand side of $f = 2c\pi \cos(2\pi x) + 4K\pi^2 \sin(2\pi x)$, $c = 10$, $K = 1$ and boundary conditions $u(0) = u(1) = 0$. The exact solution is $u(x) = \sin(2\pi x)$. Throughout the convergence test, the ratio of the domains is kept constant at $\Omega^{\mathrm{FEM}} : \Omega^{\star} : \Omega^{\mathrm{MLS}} = 6 : 1 : 6$, which may be seen in the left part of Fig. 42, where also the corresponding shape functions in the domain—according to the modified approach of [86]—are shown. The right part shows the convergence results for the two original and modified coupling approaches of [24] and [86] respectively, see sections 6.3 and 6.4. The rate of convergence remains the same as in pure FEM computations. As expected, the higher

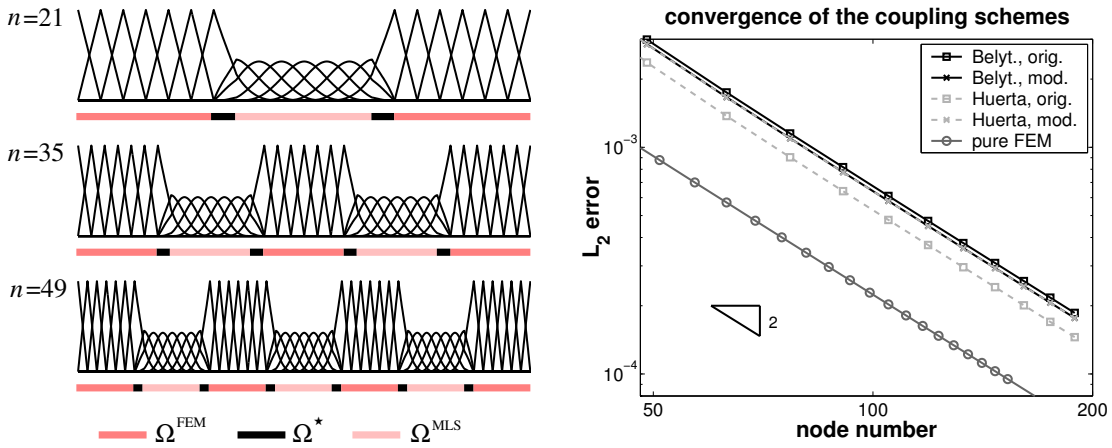

Figure 42: Convergence test of the coupling approaches.

rates of convergence often achieved for a pure MLS approximations are not observed, which is in agreement to [24, 86].

6.5.2 Driven Cavity Flow

The driven cavity test case is already described in section 5.4.2, where the approximation was made by meshfree shape functions only. Here, a coupled meshfree/meshbased discretization as shown in Fig. 43 (for 41×41 nodes) is realized. The modified coupling approaches are applied, in order to enable the use of small dilatation parameters of $\rho_i = 1.3\Delta x$. For a Reynolds number of $\mathrm{Re} = 1000$ convergence may not be reached for dilatation parameters $\rho_i \geq 2.0\Delta x$, underlining the importance of the modified coupling versions.

Details about the construction of the meshfree shape functions are given in section 4.5. Here, the MLS shape functions have rectangular supports resulting from tensor products of the weighting functions in the MLS procedure, see section 4.3.3. Integration is performed by help of a background mesh as described in section 4.3.6. It is discussed in section 4.3.5 that the imposition of essential boundary conditions requires certain attention with meshfree methods. However, this is no problem in this test case because the Dirichlet boundary conditions are applied at finite element nodes only, as may be seen from Fig. 43. If not stated otherwise, these specifications of the meshfree method hold as well for all the other test cases described in the following.

The results of both modified coupling approaches are almost identical for this test case

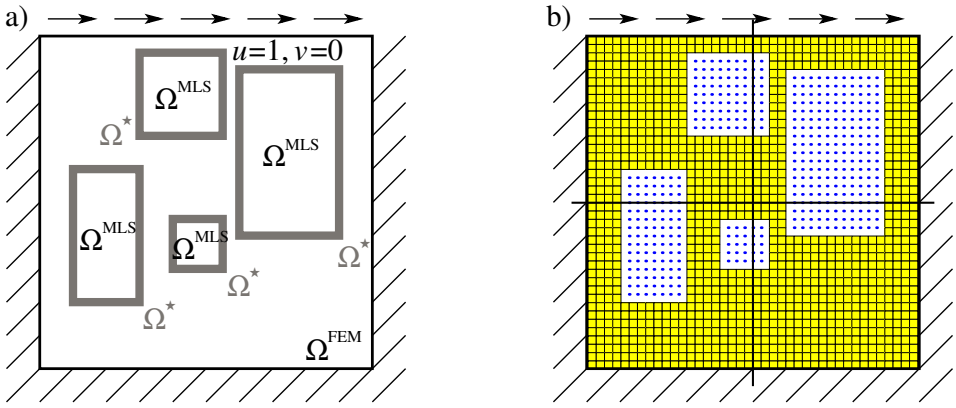

Figure 43: a) Driven cavity test case with domain decomposed into Ω^{FEM}, Ω^{MLS} and $\Omega^{\star}$, b) discretization with 41×41 FEM and MLS nodes.

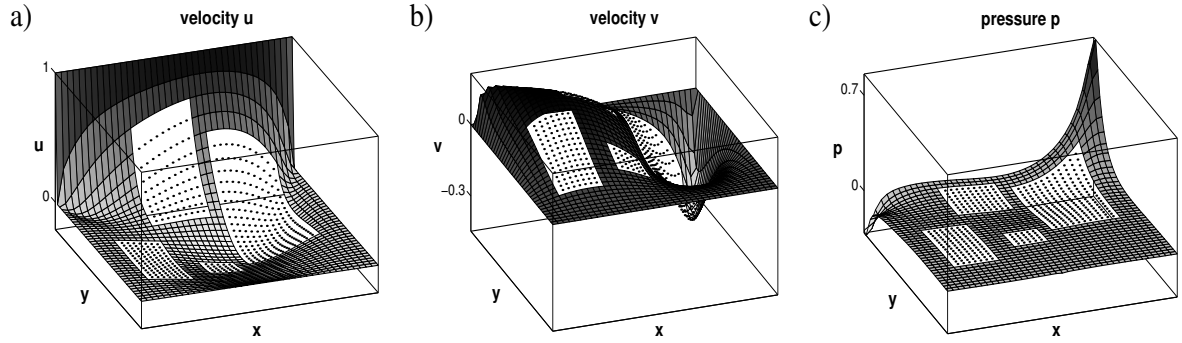

Figure 44: Solution for u, v and p over the domain Ω.

and therefore, only the result for the modified coupling approach of Huerta *et al.* [86] as proposed in section 6.3.2 is shown. Fig. 44 shows results for the unknowns u, v and p over the domain Ω; it may be seen that the meshfree and meshbased parts fit smoothly together. In Fig. 45, the convergence against the reference solution along the vertical center velocity profile is shown, and coupled results are compared with the pure FEM solutions. The Ω^{MLS} holes are placed such that the center profile directly cuts through them. In-between the MLS nodes in the figures, linear interpolation has been applied for simplicity. One may see that coupling does not adversely affect the solution.

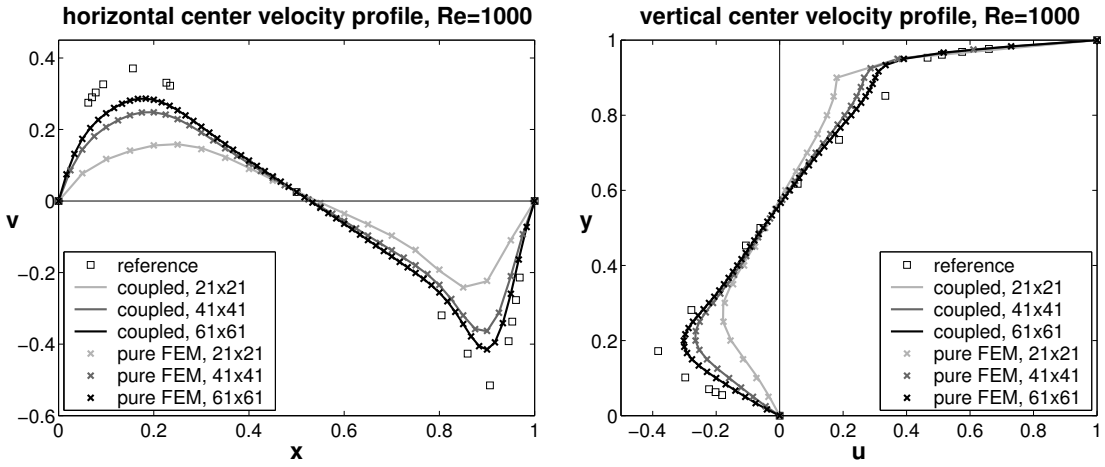

Figure 45: Velocity profiles along center cuts compare pure FEM results with the coupled results and show convergence against reference solution.

6.5.3 Cylinder Flow

The channel flow around a cylinder has been developed as a test case e.g. in [167], the setting considered here differs from the cylinder test case discussed in section 5.4.3 and [178]. The cylinder with a diameter of 0.1 is placed slightly unsymmetrically in y-direction of the channel, see Fig. 46a). No-slip boundary conditions are applied on the upper and lower wall and on the cylinder surface. A quadratic velocity profile for u, with $u_{\max} = 1.5$, and $v = 0$ are applied at the inflow on the left side of the domain. At the outflow traction-free boundary conditions are used. The density and viscosity are prescribed as $\mu = 0.001$ and $\varrho_{\mathrm{F}} = 1.0$. This results in a Reynolds number of $\mathrm{Re} = \varrho_{\mathrm{F}} \cdot u_m \cdot L/\mu = 100$ when taking the cylinder diameter as a length scale L and the average inflow velocity $u_m = 1.0$ at the inflow.

For this Reynolds number the well-known Kármán vortex street develops behind the cylinder. A quasi-stationary solution is obtained. Reference solutions are given for the lift and drag coefficients c_L and c_D of the cylinder [167]. Fig. 46 shows a sketch of the Kármán vortex street, the discretization by FEM and MLS nodes and the development of c_D and c_L in time until a periodical solution is obtained. It should be noted that the "steady-state" solution for the same Reynolds-number of section 5.4.3 is obtained by using a symmetric cylinder-setting omitting any perturbations; it is clearly not the physically relevant solution, see [178].

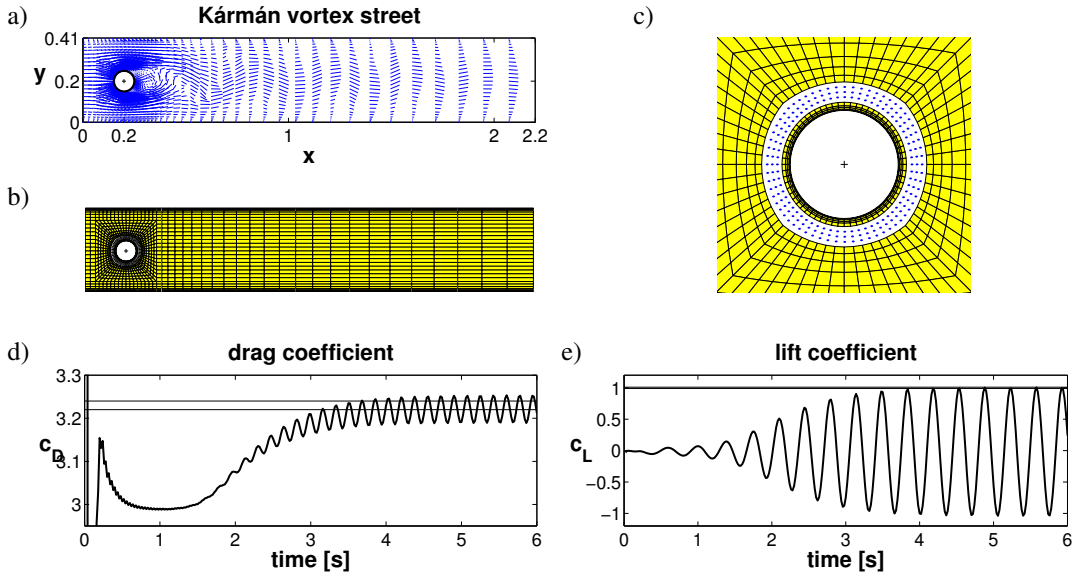

Figure 46: Cylinder test case at Re = 100, a) the Kármán vortex street, b) discretization of Ω, c) detail of the discretization with FEM and MLS nodes around the cylinder, d) and e) development of the drag and lift coefficient in time.

In the left part of Fig. 47, the results for the drag and lift coefficient obtained with the modified approaches of [24] and [86], see sections 6.3.2 and 6.4.2, are compared with the pure FEM computation ($\Delta t = 0.005$). The horizontal lines show the limits, in which the exact value for the maximum of c_D and c_L lie [167]. One may again see that the results are quite close together. The drag coefficient is slightly improved with the coupled approaches, the modified coupling approach of [86] (section 6.3) achieves somewhat better results than the approach of [24] (section 6.4). Both are slightly better than the pure FEM computation. Results for the lift coefficient are almost indistinguishable, i.e. the drag coefficient turns out to be more sensitive.

The right part of Fig. 47 shows the dependence of the drag and lift coefficient on the time step Δt. The Strouhal number $\mathrm{St} = D/(u_m T)$, with the diameter $D = 0.1$ of the cylinder, the average inflow from the left with $u_m = 1$ and the time T for 2 periods of the curve of c_D (equals 1 period of the curve of c_L), are displayed in the figure as well. A clear convergence against the reference Strouhal number of $0.295 \leq \mathrm{St} \leq 0.305$ may be seen and the amplitudes of the drag coefficient improve.

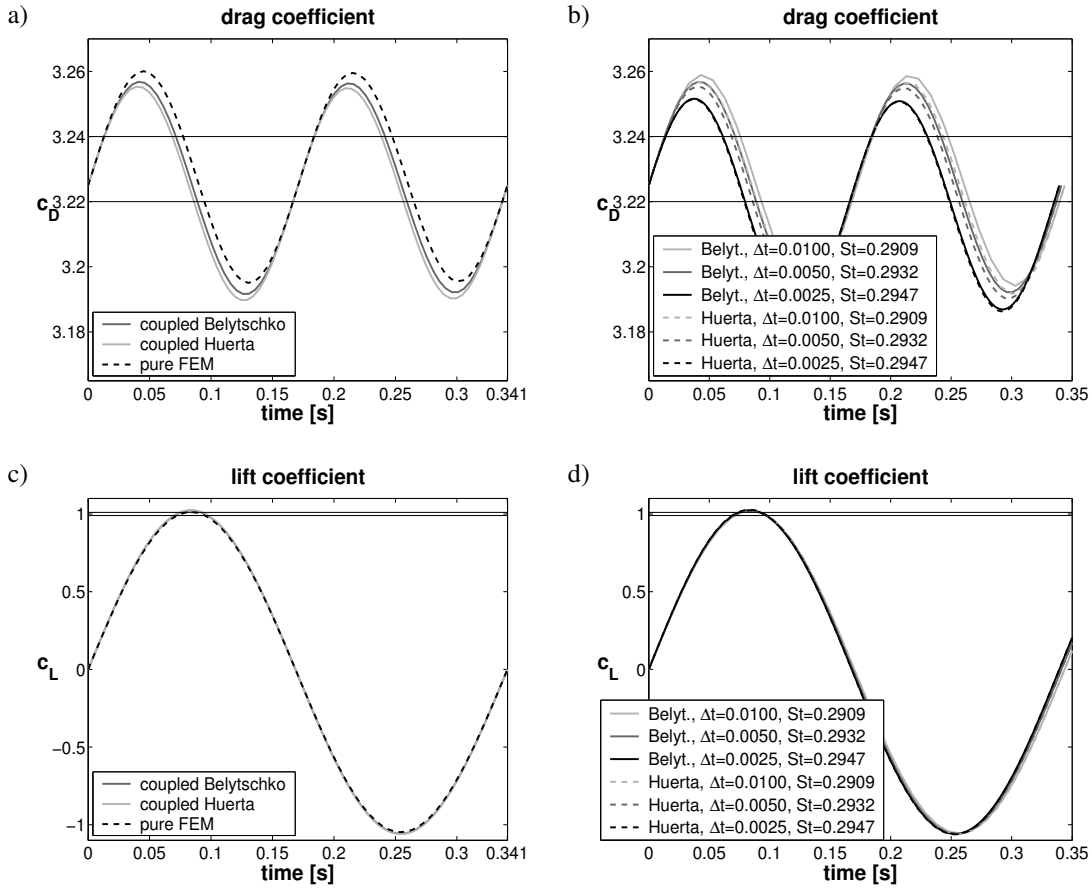

Figure 47: The left part compares the different modified coupling approaches and the pure FEM solution for $\Delta t = 0.005$, the right part shows the convergence in time of the coupled approaches.

6.5.4 Flow Around a Vortex Excited Cantilever Beam

This test case was proposed and investigated first by Wall and Ramm in [186], later from Hübner and Walhorn in [84]. A sketch of the situation is depicted in Fig. 48a). This test case is considered here in order to validate the whole procedure of simulating fluid-structure interaction phenomena as discussed in section 3.4, rather than validating only the coupled flow solver as done in the previous test cases. A square cylinder is placed in a channel flow, and a thin elastic structure is fixed at the downstream side of the cylinder. The Young's modulus of the beam-like St. Venant solid is $E = 2.5 \cdot 10^6$, Poisson's ratio is $\nu = 0.35$ and the density is $\varrho_{\mathrm{S}} = 0.1$; plane strain is considered. 2×24 bi-linear elements are used for the finite element analysis of the structure, and standard Gauss integration with 2×2 integration points is employed. Considering a linear Euler-Bernoulli beam theory, this leads to a period of $T = 0.33$ for the first eigenmode, which was confirmed by the structure solver based on the equations in section 2.2 in separate computations without the fluid. The Dirichlet boundary conditions for the structure are $d_x = d_y = 0$ along the connection with the square cylinder, and Neumann boundary conditions are the forces from the fluid acting on the structure. The Newmark algorithm was used for the time integration of the structure.

The fluid data is as follows: The fluid domain is discretized by 4128 elements and $2 \cdot 243 = 486$ MLS particles, which are coupled by the modified coupling procedure of Huerta *et al.*, as proposed in section 6.3.2. The density is $\rho_{\mathrm{F}} = 1.18 \cdot 10^{-3}$ and the viscosity is $\mu = 1.82 \cdot 10^{-4}$. No-slip boundary conditions are applied along the inner structure, i.e. at the square cylinder and the beam. Slip boundary conditions are used at the lower and upper part of the domain. The inflow velocity is $u = 51.3$, $v = 0$, and at the outflow traction-free boundary conditions are set. The Reynolds number is $\mathrm{Re} = 333$, using the width of the square cylinder as a length scale. If the beam is considered fixed, this test case reduces to a pure flow simulation, where the reference vortex shedding period of $T = 0.16$ [84] was found by the coupled flow solver as well. Therefore, it is necessary to apply an imperfection to the flow field, which was realized here by using $u(0, y) = 51.3 + y$ along the inflow until $t = 0.1$, and then using the unmodified inflow velocity of $u = 51.3$ for $t > 0.1$. The time step is chosen to be $\Delta t = 0.001$.

Considering the elasticity of the structure results in fluid-structure interaction. A strongly coupled, partitioned technique as described in section 3.4 is used for the approximation. The fluid is considered in the structure simulation by its forces acting along the fluid-

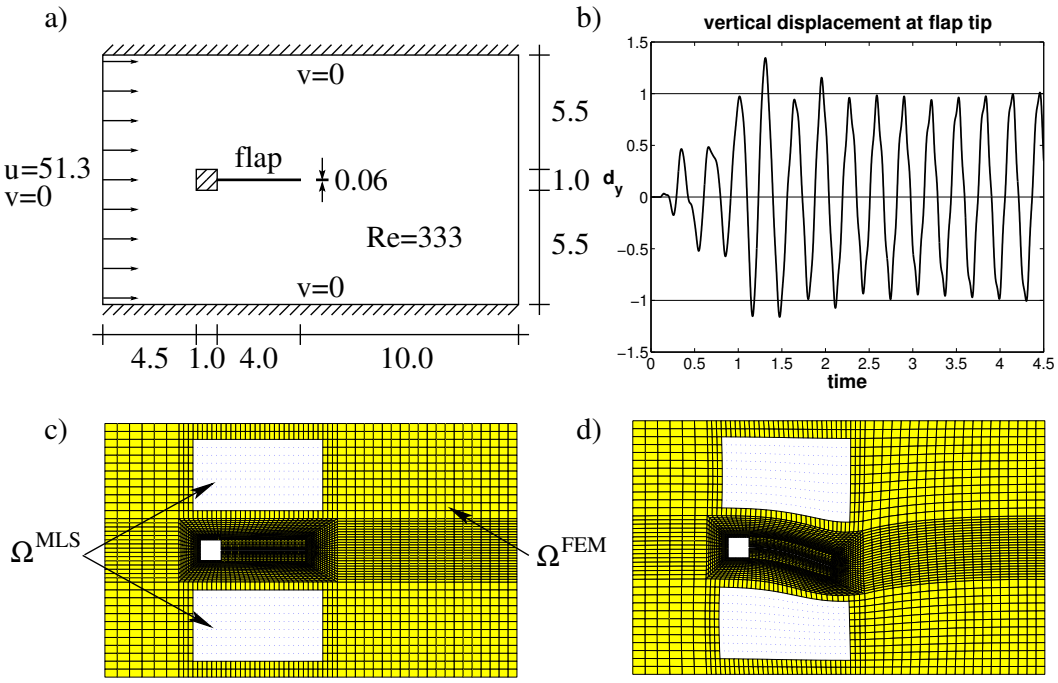

Figure 48: a) Problem statement of the vortex excited beam test case, b) vertical displacement of the beam tip, c) and d) show the discretization in the initial and deformed case.

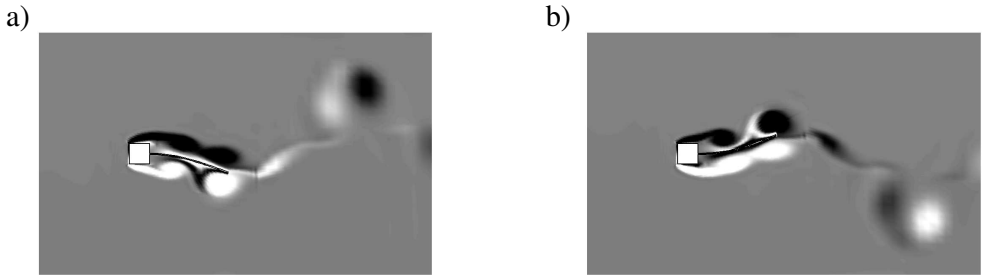

Figure 49: Vorticity for different displacements of the beam.

structure boundary, i.e. the fluid data is employed as a Neumann boundary condition for the structure. The structure is considered in the flow solver by the new position and velocity of the interface, and the velocities represent a Dirichlet boundary for the fluid problem. An important step is the update of a suitable fluid mesh, which conforms with the deformed structure mesh. This is realized by the solution of a stationary pseudo-structure problem based on linear elasticity (Hooke solid). It is well-known [13, 166] that the deformation of small elements should be limited by making them considerably stiffer than larger ones, in practice this may be realized by generating an element-size dependent Young's modulus for the pseudo-structure problem. The pseudo-structure domain equals the fluid domain in this test case, boundary conditions are of Dirichlet-type only and consist of zero-displacements along all boundaries except the interface. In Fig. 48c) and d) one may see the initial and a deformed mesh for a certain deformation of the beam. The mesh velocities resulting from the movement of the FEM and MLS nodes in the fuid domain are considered with the arbitrary Lagrangian Eulerian (ALE) technique [98].

The resulting displacement of the beam tip over time is shown in Fig. 48b). Until $t = 0.1$, where the inflow conditions are changed in order to apply an imperfection, the structure is considered fixed. Then, the vortex shedding phenomena, see Fig. 49, lead to an excitation of the beam, leading to a periodic stationary behaviour, which is established at $t \geq 4$. The motion of the beam structure shows large displacements and is dominated by the first eigenmode. The results are in excellent agreement with [84].

6.5.5 Flow Around a Rotating Obstacle

The previously described test cases verified the coupled fluid solver. However, they do not take advantage of the beneficial properties of this approach. The following test case replaces the cylinder of section 6.5.3 by a rotating obstacle. A problem statement can be seen in Fig. 50a), the corresponding discretization with the meshfree and meshbased part in b) and c). The rotation of the different rotors is prescribed and is completed after 1 time unit. The geometries of the different rotors may be seen from Fig. 50d), they are defined as follows: Each rotor wing is a segment of a circle and has a thickness of $0.5 \cdot 10^{-3}$. The inner circle of each rotor object has a diameter of $3.0 \cdot 10^{-3}$, and each rotor wing is perpendicular to the surface of this inner circle. Each rotor object has an outer radius of 0.05. Results are obtained for stiff or elastic rotors. In case of elastic material properties, Young's modulus is set to $E = 10^6$ and Poisson's ratio is $\nu = 0.3$. The HHT-α method

with $\alpha = -0.05$ has been used for time integration of the structure. The rotation of the rotors is enforced by prescribing suitable deformation for d_x and d_y at 4 nodes inside the inner circle of each rotor (at $\pm 2.0 \cdot 10^{-4}$ with respect to the center of the inner circle). The pseudo-structure domain Ω_{M} involved in the mesh-moving procedure is restricted to the inner mesh around the rotor. Thereby the boundary of the inner mesh remains circular.

The fluid density is $\varrho_{\mathrm{F}} = 1.0$ and the viscosity is $\mu = 0.001$. The number of elements in the fluid domain varies between 3316 and 4808 elements, depending on the number of rotor wings. In each case, 256 MLS nodes are coupled with the FEM domain, using the modified coupling approach of section 6.3.2 as it performs somewhat better in the previous test cases. Differing from all the other test cases in this section, the MLS shape functions have spherical instead of rectangular supports with a radius of 0.01. Furthermore, integration over individual supports is performed instead of using background meshes. The inflow and outflow boundary conditions of the flow field are identical to the cylinder test case described in section 6.5.3, no-slip boundary conditions are applied on the lower and upper wall and on the rotor surface.

For this test case, standard meshbased methods fail to give results due to the distortion of the mesh which must follow the rotation. However, this is no problem with the coupled fluid solver, where the rotating inner mesh and the stationary outer mesh are separated by a meshfree area. The mesh velocities of the inner FEM mesh and the surrounding inner MLS ring, see Fig. 50c), are considered with the arbitrary Lagrangian Eulerian (ALE) technique [98]. Fig. 51 shows vorticity results for different angles α of the two-wings rotor. In the case of only two wings, the flow field depends strongly on the angle α of the rotor, whereas for larger number of rotor wings, the flow properties (e.g. vortex shedding phenomena) become more and more similar to a cylinder test case with a diameter of 0.05.

In Fig. 52 the resulting momentum around the center of the rotors in dependence of the angle α of the inner mesh are shown for stiff rotors. Assuming the rotor to be an elastic material, Fig. 53a) shows the deformation of the rotor wings at a certain time step, b) shows the displacements—relative to the unloaded situation of the rotor undergoing a rigid body rotation—in horizontal and vertical direction of a rotor tip of the three-wing rotor over one rotation (346 bi-linear elements and 455 structure nodes are used for the finite element structural analysis of this rotor).

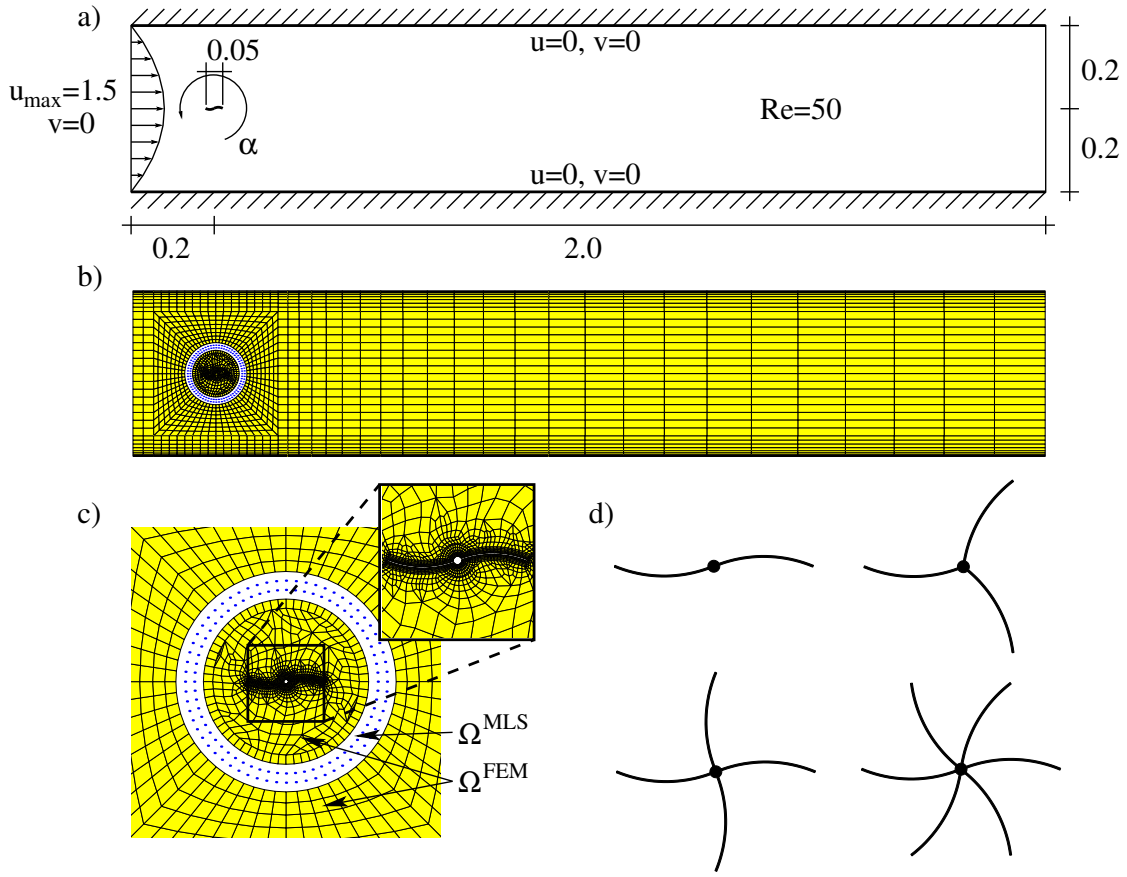

Figure 50: a) Problem statement of the rotor test case, b) and c) show the discretization and d) different rotor geometries.

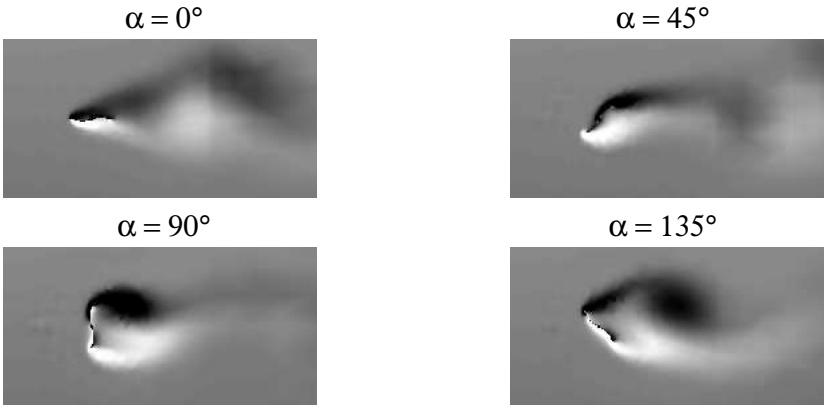

Figure 51: Vorticity for the 2-wing rotor for different angles α.

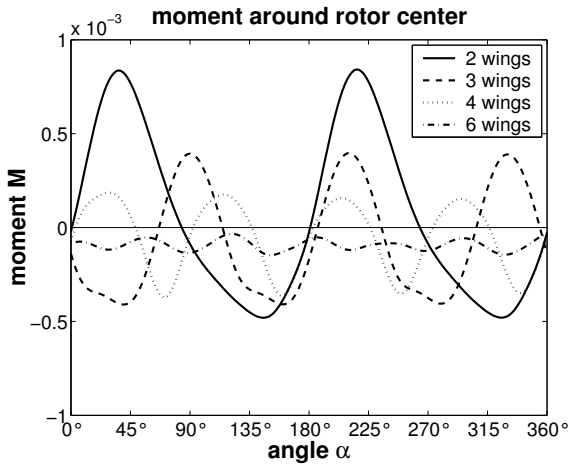

Figure 52: Resulting momentum around the center of the different rotors in dependence of the angle α.

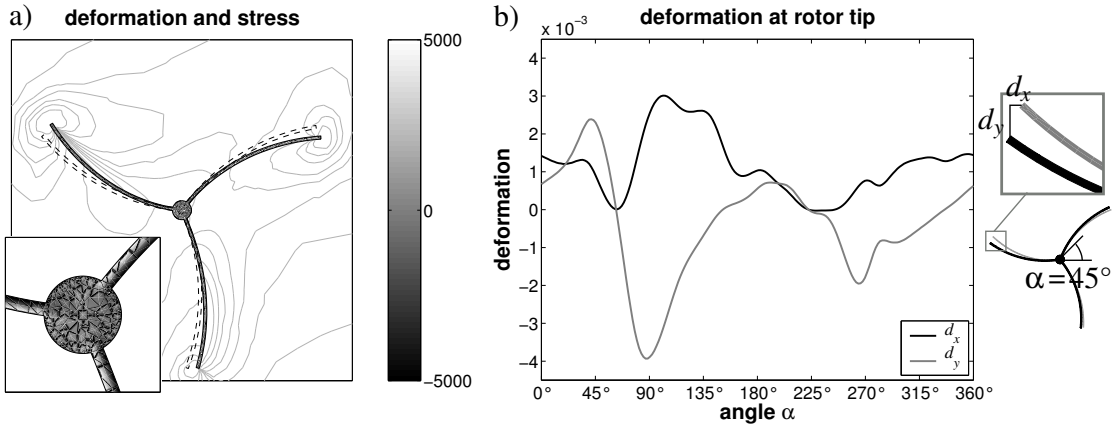

Figure 53: a) Deformation of the rotor at a certain time-step, b) distortion of a rotor tip during one rotation.

6.5.6 Flow Around a Moving Flap

This test case considers a channel flow which involves a flap undergoing large displacements. The situation is sketched in Fig. 54a). Compared to the previous test case some important differences are pointed out: The structure (flap) is surrounded by MLS nodes instead of finite elements. All finite element and MLS nodes in the fluid domain are *fixed*, a possible consideration of the large displacements of the flap by an ALE technique would fail due to the resulting distortion of the node distribution. Only the structure nodes undergo a displacement according to the flap angle α. Consequently, the nodes along the fluid-structure interface are not conforming, which has some influence in the theoretical properties of the numerical method applied for the approximation of this fluid-structure interaction problem [150]. For example, the stability and energy conservation properties change, however, this did not turn out to be critical in the present test case.

The position and movement of the structure is considered in the fluid domain by constructing the MLS shape functions according to the visibility criterion [21, 22], as discussed in section 4.3.7. Parts of the original support that are not visible from the corresponding node are neglected, leading to a reduced modified support, see also Fig. 14 on page 59. The resulting shape functions are discontinuous along the boundary of the flap. The ability of the MLS to incorporate discontinuities in this conceptionally simple way is an important advantage compared to standard meshbased methods, which require a remeshing along the flap in order to consider the discontinuity correctly.

A problem of the proposed technique is that an optimized mesh along the flap, i.e. high-aspect ratio elements in order to resolve the boundary layer, is not available. It is mentioned that this problem also exists for the frequent remeshing strategy because a fully automatic construction of high quality meshes for fluid problems is difficult to realize in practice [139].

The flap has a length of $L = 0.2$ and a thickness of $d = 2 \cdot 10^{-4}$, it is discretized by 4×30 bi-linear elements. The data for the structure is $\rho_{\mathrm{S}} = 0.1$, $E = 10^9$, and $\nu = 0.3$. The Newmark method is used for the integration in time. The time step size for the fluid and structure problem is chosen as $\Delta t = 0.01$. Dirichlet boundary conditions for the structure are prescribed displacements according to the angle α at the fixed-end near the wall. The flap angle varies from $\alpha_{\min} \leq \alpha \leq \alpha_{\max}$, with $\alpha_{\min} = 5^\circ$ and $\alpha_{\max} = 85^\circ$, following the function

$$\alpha\left(t\right) = 45^\circ + \frac{1}{2}\left(\alpha_{\max} - \alpha_{\min}\right) \cdot \sin\left(2\pi \frac{t}{t^\star}\right). \tag{6.46}$$

The frequency is defined by $t^\star$ which is set to 1 time unit.

The fluid data is defined by $\varrho_S = 1.0$ and $\mu = 0.001$, flow boundary conditions are defined as in the previous test case with no-slip conditions along the flap. As there are no fluid nodes along the flap, the fluid boundary conditions are imposed by a penalty method there, see section 4.3.5, with a penalty parameter of 10^6. Taking the length of the flap as the characteristic length, the Reynolds number for this test case is Re = 200. 1630 FEM and 735 MLS nodes are used for the approximation of the incompressible Navier-Stokes equations, see Fig. 54b) for the discretization of the fluid domain. It may be noted, that the meshfree domain compared to the whole fluid domain is considerably larger than in the prior test cases. This results in a comparably large computational effort, although the total number of nodes is only moderate.

Fig. 55 shows shape functions in the meshfree fluid domain. In 55a), the whole set of all meshfree shape functions is shown, it fulfills the conditions for first order consistency, see equation (4.1), and consequently builds a partition of unity. Along the boundaries of the meshfree domain, one may see that the coupled meshfree/meshbased shape functions look similar to standard bi-linear shape functions as they are used in the rest of the domain. It is important to note that the shape functions are discontinuous along the deformed flap. This is further stressed by displaying some selected shape functions along the flap as done in Fig. 55b) and c).

Results for this test case are displayed in Fig. 56, where the vorticity in the domain is shown for different angles α. The displacements d_x and d_y of the flap tip over one period of the flap movement are shown in the left part of Fig. 57. These displacements are with respect to the position of the flap tip at the initial flap angle of $\alpha = 45°$. The dashed line shows the displacements resulting from a rigid body motion of the flap, i.e. the displacements of the unloaded flap for certain angles α. The difference to the solid line is the deformation in consequence of the loading resulting from the fluid. The right part of Fig. 57 depicts the moment around the fixed-end of the flap over one period of the flap movement.

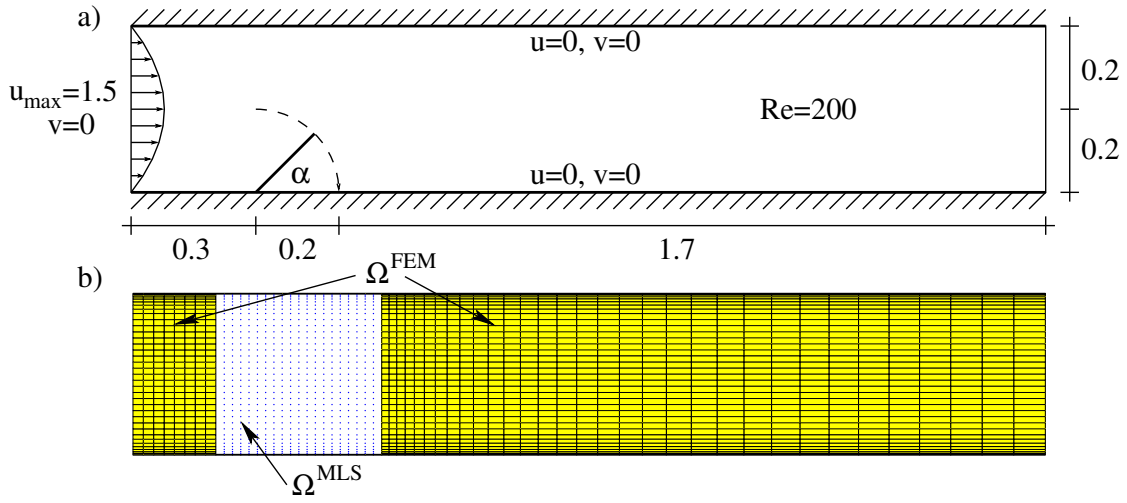

Figure 54: a) Problem statement of the flap test case, b) shows the discretization.

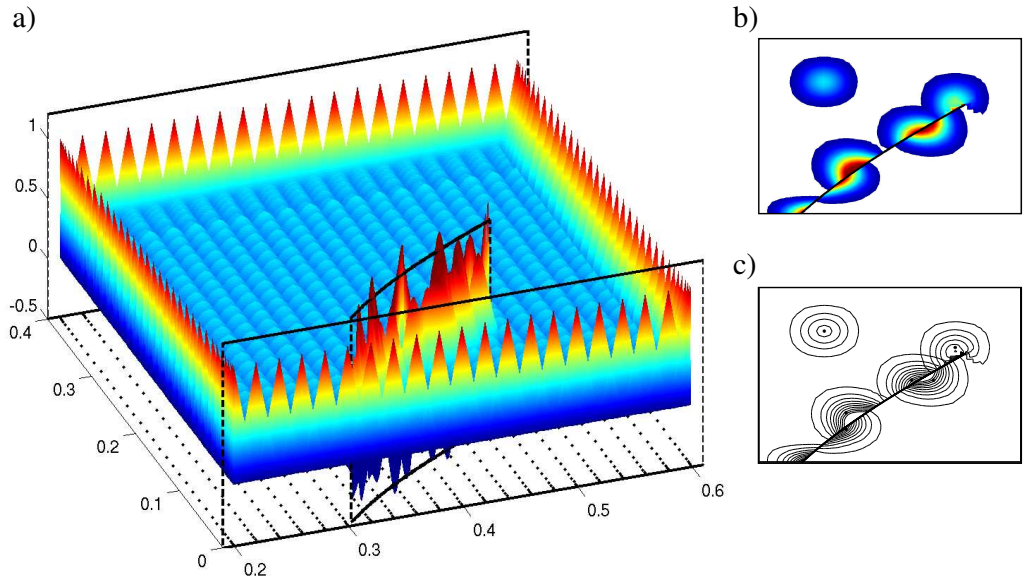

Figure 55: a) Set of all shape functions in Ω^{MLS}, b) and c) show some selected shape functions along the flap discontinuity.

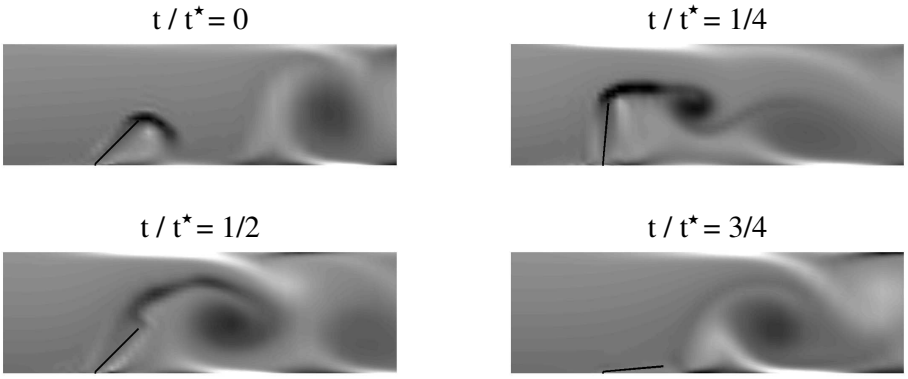

Figure 56: Vorticity of the flap test case for different angles of the flap.

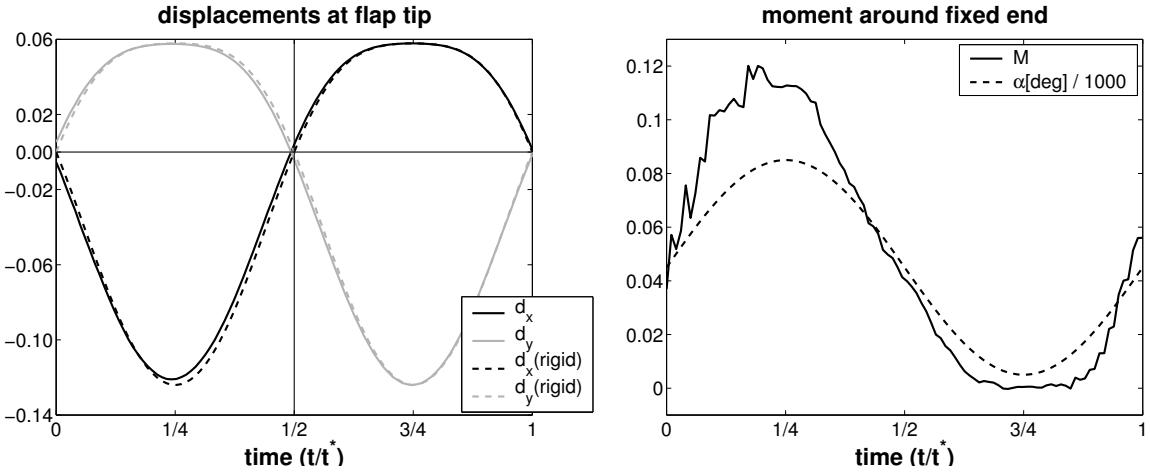

Figure 57: Displacement of the flap tip with respect to the initial flap angle of $\alpha = 45°$, resulting moment around the fixed-end of the flap.

7 Outlook and Conclusion

The aim of this work is to develop a method which is able to simulate complex flow problems—including fluid-structure interaction phenomena—, which possibly involve large geometric deformations or moving and rotating obstacles. Standard meshbased methods fail if a suitable mesh can not be maintained throughout the simulation. More advanced techniques in the meshbased context—e.g. overlapping grids, fictitious domain and boundary methods, sliding mesh and level set methods—overcome the mesh problems, each having its characteristic advantages and disadvantages. Herein, another way is proposed, which incorporates meshfree methods with their important feature that no mesh is required for the approximation.

In contrast to the common practice to employ meshfree methods in Lagrangian collocation settings, i.e. as particle methods, these methods are employed in Eulerian or ALE Galerkin formulations in this work. This formulation is standard for meshbased methods, and for a coupling of meshfree and meshbased this is the most natural choice. The coupling is desirable because meshfree Galerkin methods are considerably more time-consuming than their meshbased counterparts, but they offer a higher accuracy and robustness when compared to meshfree collocation methods.

In section 4, the background of meshfree methods is worked out in order to justify the choice of the particular method used in this work. Most importantly, the chosen method uses shape functions based on the MLS principle which are employed in Bubnov-Galerkin or certain Petrov-Galerkin formulations as thus which arise in stabilized contexts. This method is closely related to the popular EFG method.

Eulerian and ALE formulations of the equations of fluid flow involve advection terms that require stabilization. Stabilization is also one way to use equal-order-interpolation for all unknowns of the incompressible Navier-Stokes equations. The topic of stabilization is discussed in section 5, where it is found that the same structure of standard stabilization methods may be applied to meshfree methods as well, however, the question of suitable stabilization parameters requires certain attention (applicability of meshbased formulas, different versions for the support length etc.). The numerical results show that SUPG/PSPG stabilization achieves slightly better results than GLS stabilization.

Coupling is discussed in section 6. The aim is to make profit of the beneficial features of meshfree methods in complex geometry situations where conforming meshes are difficult

to maintain, and at the same time to minimize the increased computational effort involved in the meshfree approximation by using meshbased methods wherever possible. Standard coupling approaches for coupling meshfree and meshbased shape functions are described and modified such that the resulting shape functions are more suited for stabilization. It is found in the numerical results that the modified coupling approach of Huerta *et al.* [86] achieves slightly better results than the approach of Belytschko *et al.* [24]. It is mentioned that coupled formulations of higher-order consistency can be obtained straightforward. Thus, higher order convergence of the coupled meshfree/meshbased simulations may be easily achieved by using higher order finite element shape functions and enriching the basis vector in the MLS procedure.

The coupled fluid solver is validated with standard test cases and is employed for the solution of geometrically complex fluid-structure interaction problems. The conclusion is that coupled meshfree/meshbased approximations are a very promising tool for the simulation of complex fluid and fluid-structure interaction problems.

Limitations of the coupled fluid solver as proposed in this work are clearly the physically valid ranges of the employed models for the fluid and the structure (Newtonian fluid, St. Venant structure). However, conceptionally, no serious problems are expected if more eloborate fluid and structure models need to be employed.

The focus in this work is on a new concept for a numerical method, and practically relevant test cases are not in the scope of this work. Therefore, in future works with the coupled flow solver it is important to prove the usefulness of the method with a number of further test cases. Possible applications, where the features of the coupled solver are desirable, are for example simulations of biological systems. One may think of blood cells freely moving in the surrounding liquid, or the simulation of heart valves.

The efficient dynamic construction of meshfree areas in the domain (e.g. by switching off elements), employing higher orders of consistency of the coupled approximation, or making profit of the beneficial features of meshfree methods in adaptive procedures are possible extensions of the current work.

An important step towards a simulation tool for realistic problems is also the extension to three-dimensional applications. Here, the use of purely meshfree Galerkin methods is even more restricted than in two-dimensional cases, and it certainly requires another two or three generations of computers (factor 100 in performance) until a three-dimensional flow simulation with purely meshfree Galerkin methods seems realizable. However, the

computational effort in the coupled meshfree/meshbased method scales with the meshbased part, and seems much more realistic even on today's computers. Clearly, a number of other considerations have to be taken into account for a three-dimensional coupled flow solver, such as the efficient solution of the system of equations.

8 Appendix: Conventions, Symbols and Abbreviations

Conventions and Symbols

Unless convention dictates otherwise, throughout this paper normal Latin (a) or Greek (α) letters are used for scalars and scalar functions. Bold small letters ($\boldsymbol{a}$) are in general used for vectors and vector functions and bold capital letters ($\mathbf{A}$) for matrices. The following table gives a list of frequently used variables and their meaning.

symbol	meaning
$\boldsymbol{a}$	vector of unknown MLS-coefficients
$\boldsymbol{b}$	right hand side of a system of equations
c, $\boldsymbol{c}$	advection coefficients
d	number of space-dimensions
$\boldsymbol{d}$	deformation
f, $\boldsymbol{f}$	right hand side of PDE
h_{e}, h_ρ	element length, support length
i, j	integers
k	size of an intrinsic basis $\boldsymbol{p}$
l	size of an extrinsic basis $\boldsymbol{q}$
n	order of consistency
$\boldsymbol{n}$	normal vector
n_{e}	number of elements
p	pressure
$\boldsymbol{p}$	intrinsic basis
$\boldsymbol{q}$	extrinsic basis
r	node number
t	time
$\boldsymbol{u} = (u, v)$	velocities
u, u^h	function, approximated function
$\boldsymbol{u}$	vector of r nodal unknowns
$\boldsymbol{w}$	test (=weight) functions
$\boldsymbol{x}$	space coordinate

$\boldsymbol{x}_i$	position of a node (=particle, point)
$\mathbf{A}$	system of equations
$\mathbf{B}$	MLS help matrix
C^n	order of continuity
$\mathbf{D}$	matrix $D_{ij} = N_i(\boldsymbol{x}_j)$
E	Young's modulus
$\mathbf{E}$	Green-Lagrange strain tensor
$\mathbf{E}_{\text{lin}}$	linear strain tensor
$\mathbf{F}$	gradient of the deformation
$\mathbf{I}$	identity matrix
J	MLS error functional
K, $\mathbf{K}$	diffusion tensor
$\mathcal{L}$	differential operator
$\mathbf{M}$	MLS moment matrix
$\boldsymbol{N}$	shape (=trial, ansatz) functions
Pe, Re	Peclet, Reynolds number
R	ramp function
$\mathbf{S}$	second Piola-Kirchhoff stress tensor
$\mathbf{T}$	Cauchy stress tensor
$\boldsymbol{\alpha}$	multi-index
$\boldsymbol{\alpha}^i$	vector in the multi-index set $\left\{\boldsymbol{\alpha}\middle\vert\, \lvert\boldsymbol{\alpha}\rvert \leq n\right\}$
δ	Dirac-δ function, Kronecker-δ
ε	residual
ϕ	MLS weighting (=window, kernel) function
λ, η	Lamé constants
μ	dynamic viscosity
ν	Poisson's ratio
ρ	dilatation parameter (=smoothing length)
ϱ	density
$\boldsymbol{\sigma}$	linear fluid stress tensor
τ	stabilization parameter
Γ	boundary of the domain

Ω	domain

Abbreviations

The following table shows the meaning of important abbreviations used in this work.

abbr.	meaning
ALE	arbitrary Lagrangian-Eulerian
BEM	boundary element method
CSPH	corrected smoothed particle hydrodynamics
DEM	diffuse element method
EFG	element-free Galerkin
EBC	essential boundary condition
FDM	finite difference method
FEM	finite element method
FPM	finite point method
FSI	fluid-structure interaction
FVM	finite volume method
GFEM	generalized finite element method
GLS	Galerkin/least-squares
LBIE	local boundary integral equation
LSMM	least-squares meshfree method
MFEM	meshless finite element method
MFS	method of finite spheres
MLPG	meshless local Petrov-Galerkin
MLS	moving least-squares
MLSPH	moving least-squares particle hydrodynamics
MLSRK	moving least-squares reproducing kernel
MM	meshfree method
NEM	natural element method
PDE	partial differential equation
PSPG	pressure-stabilizing/Petrov-Galerkin
PU	partition of unity

PUM	partition of unity method
PUFEM	partition of unity finite element method
RKM	reproducing kernel method
RKEM	reproducing kernel element method
RKPM	reproducing kernel particle method
SPH	smoothed particle hydrodynamics
SUPG	streamline-upwind/Petrov-Galerkin
XFEM	extended finite element method

References

[1] Aliabadi, S.K.; Ray, S.E.; Tezduyar, T.E.: SUPG finite element computation of viscous compressible flows based on the conservation and entropy variables formulations. *Comput. Mech.*, **11**, 300 – 312, 1993.

[2] Aluru, N.R.: A point collocation method based on reproducing kernel approximations. *Internat. J. Numer. Methods Engrg.*, **47**, 1083 – 1121, 2000.

[3] Atluri, S.N.; Cho, J.Y.; Kim, H.-G.: Analysis of Thin Beams, Using the Meshless Local Petrov-Galerkin Method, with Generalized Moving Least Squares Interpolations. *Comput. Mech.*, **24**, 334 – 347, 1999.

[4] Atluri, S.N.; Shen, S.: *The Meshless Local Petrov-Galerkin (MLPG) Method*. Tech Science Press, Stuttgart, 2002.

[5] Atluri, S.N.; Sladek, J.; Sladek, V.; Zhu, T.: The Local Boundary Integral Equation (LBIE) and its Meshless Implementation for Linear Elasticity. *Comput. Mech.*, **25**, 180 – 198, 2000.

[6] Babuška, I.: Error-bounds for finite element method. *Numer. Math.*, **16**, 322 – 333, 1971.

[7] Babuška, I.; Banerjee, U.; Osborn, J.E.: Meshless and generalized finite element methods: A survey of some major results. In *Meshfree Methods for Partial Differential Equations*. (Griebel, M.; Schweitzer, M.A., Eds.), Vol. 26, *Lectures Notes in Computational Science and Engineering*, Springer Verlag, Berlin, 2002.

[8] Babuška, I.; Banerjee, U.; Osborn, J.E.: Survey of meshless and generalized finite element methods: A unified approach. Technical Report 02-40, TICAM, The University of Texas at Austin, 2002.

[9] Babuška, I.; Melenk, J.M.: The partition of unity finite element method. Technical Report BN-1185, Institute for Physical Science and Technology, University of Maryland, 1995.

[10] Babuška, I.; Melenk, J.M.: The Partition of Unity Method. *Internat. J. Numer. Methods Engrg.*, **40**, 727 – 758, 1997.

[11] Bakker, A.; LaRoche, R.D.; Wang, M.H.; Calabrese, R.V.: Sliding Mesh Simulation of Laminar Flow in Stirred Reactors. *Trans IChemE (Part A)*, **75**, 42 – 44, 1997.

[12] Bathe, K. J.: *Finite Element Procedures*. Prentice-Hall, Englewood Cliffs, NJ, 1996.

[13] Behr, M.; Abraham, F.: Free-Surface Flow Simulations in the Presence of Inclined Walls. *Comp. Methods Appl. Mech. Engrg.*, **191**, 5467 – 5483, 2002.

[14] Behr, M.; Tezduyar, T.: The Shear-Slip Mesh Update Method. *Comp. Methods Appl. Mech. Engrg.*, **174**, 261 – 274, 1999.

[15] Beissel, S.; Belytschko, T.: Nodal integration of the element-free Galerkin method. *Comp. Methods Appl. Mech. Engrg.*, **139**, 49 – 74, 1996.

[16] Belikov, V.V.; Ivanov, V.D.; Kontorovich, V.K.; Korytnik, S.A.; Semenov, A.Y.: The non-Sibsonian interpolation: a new method of interpolation of the values of a function on an arbitrary set of points. *Comp. Math. Math. Phys.*, **37**, 9 – 15, 1997.

[17] Belytschko, T.; Guo, Y.; Liu, W.K.; Xiao, S.P.: A unified stability analysis of meshless particle methods. *Internat. J. Numer. Methods Engrg.*, **48**, 1359 – 1400, 2000.

[18] Belytschko, T.; Hughes, T.J.R. (eds.): *Computational Methods for Transient Analysis*, Vol. 1. North-Holland, Amsterdam, 1983.

[19] Belytschko, T.; Krongauz, Y.; Dolbow, J.; Gerlach, C.: On the completeness of meshfree particle methods. *Internat. J. Numer. Methods Engrg.*, **43**, 785 – 819, 1998.

[20] Belytschko, T.; Krongauz, Y.; Fleming, M.; Organ, D.; Liu, W.K.S.: Smoothing and accelerated computations in the element free Galerkin method. *J. Comput. Appl. Math.*, **74**, 111 – 126, 1996.

[21] Belytschko, T.; Krongauz, Y.; Organ, D.; Fleming, M.; Krysl, P.: Meshless Methods: An Overview and Recent Developments. *Comp. Methods Appl. Mech. Engrg.*, **139**, 3 – 47, 1996.

[22] Belytschko, T.; Lu, Y.Y.; Gu, L.: Element-free Galerkin Methods. *Internat. J. Numer. Methods Engrg.*, **37**, 229 – 256, 1994.

[23] Belytschko, T.; Moës, N.; Usui, S.; Parimi, C.: Arbitrary discontinuities in finite elements. *Internat. J. Numer. Methods Engrg.*, **50**, 993 – 1013, 2001.

[24] Belytschko, T.; Organ, D.; Krongauz, Y.: A Coupled Finite Element–Element-free Galerkin Method. *Comput. Mech.*, **17**, 186 – 195, 1995.

[25] Belytschko, T.; Ventura, G.; Xu, J.X.: New methods for discontinuity and crack modeling in EFG. In *Meshfree Methods for Partial Differential Equations*. (Griebel, M.; Schweitzer, M.A., Eds.), Vol. 26, *Lectures Notes in Computational Science and Engineering*, Springer Verlag, Berlin, 2002.

[26] Bertrand, F.; Tanguy, P.A.; Thibault, F.: A Three-dimensional Fictitious Domain Method for Incompressible Fluid Flow Problems. *Int. J. Numer. Methods Fluids*, **25**, 719 – 736, 1997.

[27] Blom, F.J.: Considerations on the Spring Analogy. *Int. J. Numer. Methods Fluids*, **32**, 647 – 668, 2000.

[28] Bonet, J.; Kulasegaram, S.: Correction and Stabilization of Smooth Particle Hydrodynamics Methods with Applications in Metal Forming Simulations. *Internat. J. Numer. Methods Engrg.*, **47**, 1189 – 1214, 2000.

[29] Braun, J.; Sambridge, M.: A numerical method for solving partial differential equations on highly irregular evolving grids. *Nature*, **376**, 655 – 660, 1995.

[30] Brezzi, F.: On the existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers. *RAIRO Anal. Numér.*, **R-2**, 129 – 151, 1974.

[31] Brooks, A.N.; Hughes, T.J.R.: Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comp. Methods Appl. Mech. Engrg.*, **32**, 199 – 259, 1982.

[32] Chen, J.-S.; Pan, C.; Wu, C.-I.: Large Deformation Analysis of Rubber based on a Reproducing Kernel Particle Method. *Comput. Mech.*, **19**, 211 – 227, 1997.

[33] Chen, J.S.; Han, W.; You, Y.; Meng, X.: A reproducing kernel method with nodal interpolation property. *Internat. J. Numer. Methods Engrg.*, **56**, 935 – 960, 2003.

[34] Chen, J.S.; Liu, W.K. (eds.): Meshless particle methods. *Comput. Mech.*, **25**(2-3, special issue), 99 – 317, 2000.

[35] Chen, J.S.; Liu, W.K. (eds.): Meshless methods: Recent advances and new applications. *Comp. Methods Appl. Mech. Engrg.*, **193**(12-14, special issue), 933 – 1321, 2004.

[36] Chen, J.S.; Wang, H.-P.: New Boundary Condition Treatments in Meshfree Computation of Contact Problems. *Comp. Methods Appl. Mech. Engrg.*, **187**, 441 – 468, 2000.

[37] Chen, J.S.; Wu, C.T.; You, Y.: A Stabilized Conforming Nodal Integration for Galerkin Mesh-free Methods. *Internat. J. Numer. Methods Engrg.*, **50**, 435 – 466, 2001.

[38] Chen, J.S.; Yoon, S.; H.P. Wang, W.K. Liu: An improved reproducing kernel particle method for nearly incompressible finite elasticity. *Comp. Methods Appl. Mech. Engrg.*, **181**, 117 – 145, 2000.

[39] Chen, T.; Raju, I.S.: Coupling finite element and meshless local Petrov-Galerkin methods for two-dimensional potential problems. AIAA 2002-1659, NASA Langley Research Center, Hampton, USA, 2002.

[40] Chesshire, G.; Henshaw, W.D.: Composite Overlapping Meshes for the Solution of Partial Differential Equations. *J. Comput. Phys.*, **90**, 1 – 64, 1990.

[41] Choi, Y.J.; Kim, S.J.: Node Generation Scheme for the MeshFree Method by Voronoi Diagram and Weighted Bubble Packing. *Fifth U.S. National Congress on Computational Mechanics*, Boulder, CO, 1999.

[42] Christie, I.; Griffiths, D.F.; Mitchell, A.R.; Zienkiewicz, O.C.: Finite element methods for second order differential equations with significant first derivatives. *Internat. J. Numer. Methods Engrg.*, **10**, 1389 – 1396, 1976.

[43] Dilts, G.A.: Moving-Least-Squares-Particle Hydrodynamics –I. Consistency and Stability. *Internat. J. Numer. Methods Engrg.*, **44**, 1115 – 1155, 1999.

[44] Dolbow, J.; Belytschko, T.: Numerical Integration of the Galerkin Weak Form in Meshfree Methods. *Comput. Mech.*, **23**, 219 – 230, 1999.

[45] Dolbow, J.; Belytschko, T.: Volumetric locking in the element free Galerkin method. *Internat. J. Numer. Methods Engrg.*, **46**, 925 – 942, 1999.

[46] Donea, J.; Huerta, A.: *Finite Element Methods for Flow Problems*. John Wiley & Sons, Chichester, 2003.

[47] Duarte, C.A.: A review of some meshless methods to solve partial differential equations. Technical Report 95-06, TICAM, The University of Texas at Austin, 1995.

[48] Duarte, C.A.; Oden, J.T.: An h-p adaptive method using clouds. *Comp. Methods Appl. Mech. Engrg.*, **139**, 237 – 262, 1996.

[49] Duarte, C.A.M.; Oden, J.T.: H-p clouds – an h-p meshless method. *Numer. Methods Partial Differential Equations*, **12**, 673 – 705, 1996.

[50] Fernández-Méndez, S.; Huerta, A.: Coupling finite elements and particles for adaptivity. In *Meshfree Methods for Partial Differential Equations*. (Griebel, M.; Schweitzer, M.A., Eds.), Vol. 26, *Lectures Notes in Computational Science and Engineering*, Springer Verlag, Berlin, 2002.

[51] Fernández-Méndez, S.; Huerta, A.: Imposing essential boundary conditions in meshfree methods. *Comp. Methods Appl. Mech. Engrg.*, **193**, 1257 – 1275, 2004.

[52] Ferziger, J.H.; Peric, M.: *Computational Methods for Fluid Dynamics*. Springer, Berlin, 2002.

[53] Franca, L.P.; Frey, S.L.: Stabilized finite element methods: II. The incompressible Navier-Stokes equations. *Comp. Methods Appl. Mech. Engrg.*, **99**, 209 – 233, 1992.

[54] Franca, L.P.; Frey, S.L.; Hughes, T.J.R.: Stabilized finite element methods: I. Application to the advective-diffusive model. *Comp. Methods Appl. Mech. Engrg.*, **95**, 253 – 276, 1992.

[55] Franca, L.P.; Hughes, T.J.R.: Two classes of mixed finite element methods. *Comp. Methods Appl. Mech. Engrg.*, **69**, 89 – 129, 1988.

[56] Fries, T.P.; Matthies, H.G.: Classification and Overview of Meshfree Methods. Informatikbericht-Nr. 2003-03, Technical University Braunschweig, (http://opus.tu-bs.de/opus/volltexte/2003/418/), Brunswick, 2003.

[57] Fries, T.P.; Matthies, H.G.: SUPG Stabilization of Meshfree Methods for Convection-dominated Problems. *Proceedings in Applied Mathematics and Mechanics*, Vol. 3, John Wiley & Sons, Chichester, 2003.

[58] Fries, T.P.; Matthies, H.G.: Coupled Meshfree/Meshbased Methods for Fluid Dynamics: An Alternative to the Chimera Grid Technique. *Proceedings of the XXXII International Summer School - Conference: Advanced Problems in Mechanics (APM 2004)*, St. Petersburg, Russia, 2004.

[59] Fries, T.P.; Matthies, H.G.: Coupling Meshfree Methods and FEM for the Solution of the Incompressible Navier-Stokes Equations. *Proceedings in Applied Mathematics and Mechanics*, Vol. 4, John Wiley & Sons, Chichester, 2004.

[60] Fries, T.P.; Matthies, H.G.: A Review of Petrov-Galerkin Stabilization Approaches and an Extension to Meshfree Methods. Informatikbericht-Nr. 2004-01, Technical University of Braunschweig, (http://opus.tu-bs.de/opus/volltexte/2004/549/), Brunswick, 2004.

[61] Fries, T.P.; Matthies, H.G.: Stabilized and Coupled FEM/EFG Approximations for Fluid Problems. *Proceedings of the Sixth World Congress on Computational Mechanics (WCCM VI)*, Beijing, China, 2004.

[62] Fries, T.P.; Matthies, H.G.: A Coupled Meshfree/Meshbased Method for Complex Fluid-Structure Interaction Problems. In *Proceedings of the ECCOMAS Thematic Conference on Coupled Problems*. (M. Papadrakakis, E. Oñate; Schrefler, B., Eds.), Santorini, Greece, 2005.

[63] Fries, T.P.; Matthies, H.G.: Meshfree Petrov-Galerkin Methods for the Incompressible Navier-Stokes Equations. In *Meshfree Methods for Partial Differential Equations*. (Griebel, M.; Schweitzer, M.A., Eds.), Vol. 43, *Lectures Notes in Computational Science and Engineering*, Springer Verlag, Berlin, 2005.

[64] Fries, T.P.; Matthies, H.G.: Simulation of Incompressible Flows with Coupled FEM/EFG Shape Functions. *Proceedings of the GAMM (Gesellschaft für Angewandte Mathematik und Mechanik) 2005*, Luxemburg, 2005.

[65] Fries, T.P.; Matthies, H.G.: A Stabilized and Coupled Meshfree/Meshbased Method for the Incompressible Navier-Stokes Equations — Part I: Stabilization. Informatikbericht-Nr. 2005-02, Technical University of Braunschweig, (http://opus.tu-bs.de/opus/volltexte/2005/677/), Brunswick, 2005.

[66] Fries, T.P.; Matthies, H.G.: A Stabilized and Coupled Meshfree/Meshbased Method for the Incompressible Navier-Stokes Equations — Part II: Coupling. Informatikbericht-Nr. 2005-03, Technical University of Braunschweig, (http://opus.tu-bs.de/opus/volltexte/2005/678/), Brunswick, 2005.

[67] Ghia, U.; Ghia, K.N.; Shin, C.T.: High-Re solutions for incompressible flow using the Navier-Stokes equations and a multi-grid method. *J. Comput. Phys.*, **48**, 387 – 411, 1982.

[68] Gingold, R.A.; Monaghan, J.J.: Kernel Estimates as a Basis for General Particle Methods in Hydrodynamics. *J. Comput. Phys.*, **46**, 429 – 453, 1982.

[69] Glowinski, R.; Pan, T.W.; Hesla, T.I.; Joseph, D.D.; Périaux, J.: A Distributed Lagrange Multiplier/Fictitious Domain Method for Flows around Moving Rigid Bodies: Application to Particulate Flow. *Int. J. Numer. Methods Fluids*, **30**, 1043 – 1066, 1999.

[70] Gresho, P.M.; Sani, R.L.: *Incompressible Flow and the Finite Element Method*, Vol. 1+2. John Wiley & Sons, Chichester, 2000.

[71] Griebel, M.; Schweitzer, M.A.: A particle-partition of unity method–Part II: Efficient cover construction and reliable integration. *SIAM J. Sci. Comput.*, **23**(5), 1655 – 1682, 2002.

[72] Griebel, M.; Schweitzer, M.A. (eds.): *Meshfree Methods for Partial Differential Equations*, Vol. 26, *Lectures Notes in Computational Science and Engineering*. Springer Verlag, Berlin, 2002.

[73] Griebel, M.; Schweitzer, M.A. (eds.): *Meshfree Methods for Partial Differential Equations II*, Vol. 43, *Lectures Notes in Computational Science and Engineering*. Springer Verlag, Berlin, 2005.

[74] Güler, I.; Behr, M.; Tezduyar, T.: Parallel Finite Element Computation of Free-Surface Flows. *Comput. Mech.*, **23**, 117 – 123, 1999.

[75] Günther, F.C.: *A Meshfree Formulation for the Numerical Solution of the Viscous Compressible Navier-Stokes Equations*. Phd-Dissertation, Northwestern University, Evanston, IL, 1998.

[76] Han, W.; Meng, X.: Error analysis of the reproducing kernel particle method. *Comp. Methods Appl. Mech. Engrg.*, **190**, 6157 – 6181, 2001.

[77] Han, W.; Meng, X.: Some studies of the reproducing kernel particle method. In *Meshfree Methods for Partial Differential Equations.* (Griebel, M.; Schweitzer, M.A., Eds.), Vol. 26, *Lectures Notes in Computational Science and Engineering*, Springer Verlag, Berlin, 2002.

[78] Han, X.; Oliveira, S.; Stewart, D.: Finding Sets Covering a Point with Applications to Mesh-free Galerkin Methods. *SIAM J. Comput.*, **30**, 1368 – 1383, 2000.

[79] Hegen, D.: Element-free Galerkin Methods in Combination with Finite Element Approaches. *Comp. Methods Appl. Mech. Engrg.*, **135**, 143 – 166, 1996.

[80] Heinrich, J.C.; Huyakorn, P.S.; Zienkiewicz, O.C.; Mitchell, A.R.: An 'upwind' finite element scheme for two-dimensional convective transport equation. *Internat. J. Numer. Methods Engrg.*, **11**, 131 – 143, 1977.

[81] Hilber, H.M.; Hughes, T.J.R.; Taylor, R.L.: Improved Numerical Dissipation for Time Integration Algorithms in Structural Dynamics. *Earthquake Engineering and Structural Dynamics*, **5**, 283 – 292, 1977.

[82] Hirsch, C.: *Numerical Computation of Internal and External Flows*, Vol. 1 – 2. John Wiley & Sons, Chichester, 1988.

[83] Houzeaux, G.; Codina, R.: A Chimera Method Based on a Dirichlet/Neumann(Robin) Coupling for the Navier-Stokes Equations. *Comp. Methods Appl. Mech. Engrg.*, **192**, 3343 – 3377, 2003.

[84] Hübner, B.; Walhorn, E.; Dinkler, D.: Strongly Coupled Analysis of Fluid-Structure Interaction Using Space-Time Elements. *European Conference on Computational Mechanics (ECCM) 2001*, Cracow, Poland, 2001.

[85] Huerta, A.; Fernández-Méndez, S.: Coupling Element Free Galerkin and Finite Element Methods. *ECCOMAS 2000*, CIMNE, Barcelona, 11.-14. September 2000.

[86] Huerta, A.; Fernández-Méndez, S.: Enrichment and Coupling of the Finite Element and Meshless Methods. *Internat. J. Numer. Methods Engrg.*, **48**, 1615 – 1636, 2000.

[87] Huerta, A.; Fernández-Méndez, S.: Locking in the incompressible limit for the element free Galerkin method. *Internat. J. Numer. Methods Engrg.*, **50**, 1 – 23, 2001.

[88] Huerta, A.; Fernández-Méndez, S.; Liu, W.K.: A comparison of two formulations to blend finite elements and mesh-free methods. *Comp. Methods Appl. Mech. Engrg.*, **193**, 1105 – 1117, 2004.

[89] Huerta, A.; Fernández-Méndez, S.M.: Time accurate consistently stabilized mesh-free methods for convection-dominated problems. *Internat. J. Numer. Methods Engrg.*, **50**, 1 – 18, 2001.

[90] Huerta, A.; Vidal, Y.; Villon, P.: Pseudo-divergence-free element free Galerkin method for incompressible fluid flow. *Comp. Methods Appl. Mech. Engrg.*, **193**, 1119 – 1136, 2004.

[91] Hughes, T.J.R.: A simple scheme for developing 'upwind' finite elements. *Internat. J. Numer. Methods Engrg.*, **12**, 1359 – 1365, 1978.

[92] Hughes, T.J.R.: *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, Englewood Cliffs, NJ, 1987.

[93] Hughes, T.J.R.: Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. *Comp. Methods Appl. Mech. Engrg.*, **127**, 387 – 401, 1995.

[94] Hughes, T.J.R.; Brooks, A.N.: A multidimensional upwind scheme with no crosswind diffusion. In *ASME Monograph AMD-34*. (Hughes, T.J.R., Ed.), Vol. 34, ASME, New York, NY, 1979.

[95] Hughes, T.J.R.; Franca, L.P.: A new finite element formulation for computational fluid dynamics: VII. The Stokes problem with various well-posed boundary conditions: symmetric formulations that converge for all velocity/pressure spaces. *Comp. Methods Appl. Mech. Engrg.*, **65**, 85 – 96, 1987.

[96] Hughes, T.J.R.; Franca, L.P.; Balestra, M.: A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška-Brezzi condition: a stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations. *Comp. Methods Appl. Mech. Engrg.*, **59**, 85 – 99, 1986.

[97] Hughes, T.J.R.; Franca, L.P.; Hulbert, G.M.: A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/Least-squares method for advective-diffusive equations. *Comp. Methods Appl. Mech. Engrg.*, **73**, 173 – 189, 1989.

[98] Hughes, T.J.R.; Liu, W.K.; Zimmermann, T.K.: Lagrangian-Eulerian Finite Element Formulation for Incompressible Viscous Flows. *Comp. Methods Appl. Mech. Engrg.*, **29**, 329 – 349, 1981.

[99] Hughes, T.J.R.; Mallet, M.: A new finite element formulation for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective-diffusive systems. *Comp. Methods Appl. Mech. Engrg.*, **58**, 305 – 328, 1986.

[100] Hughes, T.J.R.; Mallet, M.: A new finite element formulation for computational fluid dynamics: IV. A discontinuity-capturing operator for multidimensional advective-diffusive systems. *Comp. Methods Appl. Mech. Engrg.*, **58**, 329 – 336, 1986.

[101] Hughes, T.J.R.; Mallet, M.; Mizukami, A.: A new finite element formulation for computational fluid dynamics: II. Beyond SUPG. *Comp. Methods Appl. Mech. Engrg.*, **54**, 341 – 355, 1986.

[102] Hughes, T.J.R.; Tezduyar, T.E.: Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations. *Comp. Methods Appl. Mech. Engrg.*, **45**, 217 – 284, 1984.

[103] Idelsohn, S.R.; Oñate, E.; Calvo, N.; Del Pin, F.: Meshless finite element ideas. *Proceedings of the Fifth World Congress on Computational Mechanics (WCCM V)*, Vienna, 2002.

[104] Idelsohn, S.R.; Oñate, E.; Calvo, N.; Pin, F. Del: The meshless finite element method. *Internat. J. Numer. Methods Engrg.*, **58**, 893 – 912, 2003.

[105] Jiang, B.N.: *The least-squares finite element method—theory and applications in computational fluid dynamics and electromagnetics*. Springer Verlag, Berlin, 1998.

[106] Jin, X.; Li, G.; Aluru, N.R.: Positivity conditions in meshless collocation methods. *Comp. Methods Appl. Mech. Engrg.*, **193**, 1171 – 1202, 2004.

[107] Johnson, C.; Nävert, U.; Pitkäranta, J.: Finite element methods for linear hyperbolic problems. *Comp. Methods Appl. Mech. Engrg.*, **45**, 285 – 312, 1984.

[108] Johnson, G.R.; Stryk, R.A.; Beissel, S.R.: SPH for high velocity impact computations. *Comp. Methods Appl. Mech. Engrg.*, **139**, 347 – 373, 1996.

[109] Kaljevic, I.; Saigal, S.: An improved element free Galerkin formulation. *Internat. J. Numer. Methods Engrg.*, **40**, 2953 – 2974, 1997.

[110] Kelly, D.W.; Nakazawa, S.; Zienkiewicz, O.C.: A note on upwinding and anisotriopic balancing dissipation in finite element approximations to convective diffusion problems. *Internat. J. Numer. Methods Engrg.*, **15**, 1705 – 1711, 1980.

[111] Klaas, O.; Shepard, M.S.: An Octree Based Partition of Unity Method for Three Dimensional Problems. *Fifth U.S. National Congress on Computational Mechanics*, Boulder, CO, 1999.

[112] Koshizuka, S.; Nobe, A.; Oka, Y.: Numerical analysis of breaking waves using moving particle semi-implicit method. *Int. J. Numer. Methods Fluids*, **26**, 751 – 769, 1998.

[113] Krongauz, Y.; Belytschko, T.: Enforcement of Essential Boundary Conditions in Meshless Approximations Using Finite Elements. *Comp. Methods Appl. Mech. Engrg.*, **131**, 133 – 145, 1996.

[114] Krongauz, Y.; Belytschko, T.: A Petrov-Galerkin Diffuse Element Method (PG DEM) and its comparison to EFG. *Comput. Mech.*, **19**, 327 – 333, 1997.

[115] Krongauz, Y.; Belytschko, T.: EFG approximations with discontiuous derivatives. *Internat. J. Numer. Methods Engrg.*, **41**, 1215 – 1233, 1998.

[116] Krysl, P.; Belytschko, T.: Element-free Galerkin method: Convergence of the continuous and discontinuous shape functions. *Comp. Methods Appl. Mech. Engrg.*, **148**, 257 – 277, 1997.

[117] Krysl, P.; Belytschko, T.: An Efficient Linear-precision Partition of Unity Basis for Unstructured Meshless Methods. *Commun. Numer. Meth. Engrg.*, **16**, 239 – 255, 2000.

[118] Krysl, P.; Belytschko, T.: ESFLIB: A Library to Compute the Element Free Galerkin Shape Functions. *Comp. Methods Appl. Mech. Engrg.*, **190**, 2181 – 2205, 2001.

[119] Kuhnert, J.: An upwind finite pointset method (FPM) for compressible Euler and Navier-Stokes equations. In *Meshfree Methods for Partial Differential Equations.*

(Griebel, M.; Schweitzer, M.A., Eds.), Vol. 26, *Lectures Notes in Computational Science and Engineering*, Springer Verlag, Berlin, 2002.

[120] Kulasegaram, S.; Bonet, J.; Lok, T.-S.L.; Rodriguez-Paz, M.: Corrected Smooth Particle Hydrodynamics - A Meshless Method for Computational Mechanics. *EC-COMAS 2000*, CIMNE, Barcelona, 11.-14. September 2000.

[121] Lancaster, P.; Salkauskas, K.: Surfaces Generated by Moving Least Squares Methods. *Math. Comput.*, **37**, 141 – 158, 1981.

[122] Li, S.; Liu, W.K.: Moving least-squares reproducing kernel method Part II: Fourier analysis. *Comp. Methods Appl. Mech. Engrg.*, **139**, 159 – 193, 1996.

[123] Li, S.; Liu, W.K.: Reproducing Kernel Hierarchical Partition of Unity, Part I – Formulation and Theory. *Internat. J. Numer. Methods Engrg.*, **45**, 251 – 288, 1999.

[124] Li, S.; Liu, W.K.: Reproducing Kernel Hierarchical Partition of Unity, Part II – Applications. *Internat. J. Numer. Methods Engrg.*, **45**, 289 – 317, 1999.

[125] Li, S.; Liu, W.K.: Meshfree and particle methods and their applications. *Appl. Mech. Rev.*, **55**, 1 – 34, 2002.

[126] Li, S.; Liu, W.K.: *Meshfree Particle Methods*. Springer, Berlin, 2004.

[127] Li, S.; Lu, H.; Han, W.; Liu, W.K.; Simkins, D.C.: Reproducing kernel element method. Part II: Globally conforming I^m/C^n hierarchies. *Comp. Methods Appl. Mech. Engrg.*, **193**, 953 – 987, 2004.

[128] Li, X.-Y.; S.-H.Teng; Üngör, A.: Point Placement for Meshless Methods using Sphere Packing and Advancing Front Methods. *ICCES'00*, Los Angeles, USA, 20.-25. August 2000.

[129] Li, X.-Y.; S.-H.Teng; Üngör, A.: Biting: Advancing Front Meets Sphere Packing. *Internat. J. Numer. Methods Engrg.*, **49**, 61 – 81, 2000.

[130] Liu, G.R.: *Meshless Methods*. CRC Press, Boca Raton, 2002.

[131] Liu, G.R.; Gu, T.: Meshless local Petrov-Galerkin (MLPG) method in combination with finite element and boundary element approaches. *Comput. Mech.*, **26**, 536 – 546, 2000.

[132] Liu, W.K.; Belytschko, T.; Oden, J.T. (eds.): Meshless methods. *Comp. Methods Appl. Mech. Engrg.*, **139**(1-4, special issue), 1 – 400, 1996.

[133] Liu, W.K.; Chen, Y.: Wavelet and Multiple Scale Reproducing Kernel Methods. *Int. J. Numer. Methods Fluids*, **21**, 901 – 931, 1995.

[134] Liu, W.K.; Chen, Y.; Uras, R.A.; Chang, C.T.: Generalized Multiple Scale Reproducing Kernel Particle Methods. *Comp. Methods Appl. Mech. Engrg.*, **139**, 91 – 157, 1996.

[135] Liu, W.K.; Han, W.; Lu, H.; Li, S.; Cao, J.: Reproducing kernel element method. Part I: Theoretical formulation. *Comp. Methods Appl. Mech. Engrg.*, **193**, 933 – 951, 2004.

[136] Liu, W.K.; Jun, S.; Zhang, Y.F.: Reproducing Kernel Particle Methods. *Int. J. Numer. Methods Fluids*, **20**, 1081 – 1106, 1995.

[137] Liu, W.K.; Li, S.; Belytschko, T.: Moving Least Square Reproducing Kernel Methods (I) Methodology and Convergence. *Comp. Methods Appl. Mech. Engrg.*, **143**, 113 – 154, 1997.

[138] Liu, W.K.; Uras, R.A.; Chen, Y.: Enrichment of the finite element method with the reproducing kernel particle method. *J. Appl. Mech., ASME*, **64**, 861 – 870, 1997.

[139] Löhner, R.: *Applied CFD Techniques*. John Wiley & Sons, Chichester, 2001.

[140] Lu, H.; Li, S.; Simkins, D.C.; Liu, W.K.; Cao, J.: Reproducing kernel element method. Part III: Generalized enrichment and application. *Comp. Methods Appl. Mech. Engrg.*, **193**, 989 – 1011, 2004.

[141] Lu, Y.Y.; Belytschko, T.; Gu, L.: A New Implementation of the Element Free Galerkin Method. *Comp. Methods Appl. Mech. Engrg.*, **113**, 397 – 414, 1994.

[142] Lucy, L.B.: A numerical approach to the testing of the fission thesis. *Astronom. J.*, **82**(12), 1013 – 1024, 1977.

[143] Masud, A.; Hughes, T.J.R.: A Space-Time Galerkin/Least-Squares Finite Element Formulation of the Navier-Stokes Equations for Moving Domain Problems. *Comp. Methods Appl. Mech. Engrg.*, **146**, 91 – 126, 1997.

[144] Matthies, H.G.; Steindorf, J.: Partioned Strong Coupling Algorithms for Fluid-Structure Interaction. *Computers & Structures*, **81**, 805 – 812, 2003.

[145] Melenk, J.M.; Babuška, I.: The Partition of Unity Finite Element Method: Basic Theory and Applications. *Comp. Methods Appl. Mech. Engrg.*, **139**, 289 – 314, 1996.

[146] Mittal, S.: On the performance of high aspect ratio elements for incompressible flows. *Comp. Methods Appl. Mech. Engrg.*, **188**, 269 – 287, 2000.

[147] Mizukami, A.; Hughes, T.J.R.: A Petrov-Galerkin finite element method for convection-dominated flows: an accurate upwinding technique for satisfying the maximum principle. *Comp. Methods Appl. Mech. Engrg.*, **50**, 181 – 193, 1985.

[148] Monaghan, J.J.: Why particle methods work. *SIAM J. Sci. Comput.*, **3**, 422 – 433, 1982.

[149] Monaghan, J.J.: An introduction to SPH. *Comput. Phys. Comm.*, **48**, 89 – 96, 1988.

[150] Morand, H.J.P.; Ohayon, R.: *Fluid-Structure Interaction*. John Wiley & Sons, Chichester, 1995.

[151] Mukherjee, Y.X.; Mukherjee, S.: On Boundary Conditions in the Element-free Galerkin Method. *Comput. Mech.*, **19**, 264 – 270, 1997.

[152] Muravin, B.; Turkel, E.: Advance diffraction method as a tool for solution of complex non-convex boundary problems. In *Meshfree Methods for Partial Differential Equations*. (Griebel, M.; Schweitzer, M.A., Eds.), Vol. 26, *Lectures Notes in Computational Science and Engineering*, Springer Verlag, Berlin, 2002.

[153] Nävert, U.: *Finite element methods for convection-diffusion problems*. Dissertation, Dep. of Computer Science, Chalmers University of Technology, Göteborg, Sweden, 1982.

[154] Nayroles, B.; Touzot, G.; Villon, P.: Generalizing the Finite Element Method: Diffuse Approximation and Diffuse Elements. *Comput. Mech.*, **10**, 307 – 318, 1992.

[155] Newmark, N.M.: A Method of Computation for Structural Dynamics. *J. Engng. Mech. Div. Proc. ASCE*, **85**, 67 – 94, 1959.

[156] Noguchi, H.; Kawashima, T.; Miyamura, T.: Element free analyses of shell and spatial structures. *Internat. J. Numer. Methods Engrg.*, **47**, 1215 – 1240, 2000.

[157] Oden, J.T.; Duarte, C.A.; Zienkiewicz, O.C.: A New Cloud-based hp Finite Element Method. *Comp. Methods Appl. Mech. Engrg.*, **153**, 117 – 126, 1998.

[158] Oñate, E.; Idelsohn, S.; Zienkiewicz, O.C.; Taylor, R.L.: A Finite Point Method in Computational Mechanics. Applications to Convective Transport and Fluid Flow. *Internat. J. Numer. Methods Engrg.*, **39**, 3839 – 3866, 1996.

[159] Oñate, E.; Idelsohn, S.; Zienkiewicz, O.C.; Taylor, R.L.; Sacco, C.: A Stabilized Finite Point Method for Analysis of Fluid Mechanics Problems. *Comp. Methods Appl. Mech. Engrg.*, **139**, 315 – 346, 1996.

[160] Organ, D.; Fleming, M.; Terry, T.; Belytschko, T.: Continous meshless approximations for nonconvex bodies by diffraction and transparency. *Comput. Mech.*, **18**, 225 – 235, 1996.

[161] Osher, S.; Fedkiw, R.P.: *Level Set Methods and Dynamic Implicit Surfaces*. Springer Verlag, Berlin, 2003.

[162] Quarteroni, A.; Tuveri, M.; Veneziani, A.: Computational Vascular Fluid Dynamics: Problems, Models and Methods. *Comput. Visual Sci.*, **2**, 163 – 197, 2000.

[163] Rabczuk, T.; Belytschko, T.; Xiao, S.P.: Stable particle methods based on Lagrangian kernels. *Comp. Methods Appl. Mech. Engrg.*, **193**, 1035 – 1063, 2004.

[164] Randles, P.W.; Libersky, L.D.: Smoothed Particle Hydrodynamics: Some recent improvements and applications. *Comp. Methods Appl. Mech. Engrg.*, **139**, 375 – 408, 1996.

[165] Rice, J.G.; Schnipke, R.J.: A monotone streamline upwind finite element method for convection-dominated flows. *Comp. Methods Appl. Mech. Engrg.*, **48**, 313 – 327, 1985.

[166] Sackinger, P.A.; Schunk, P.R.; Rao, R.R.: A Newton-Raphson Pseudo-Solid Domain Mapping Technique for Free and Moving Boundary Problems: A Finite Element Implementation. *J. Comput. Phys.*, **125**, 83 – 103, 1996.

[167] Schäfer, M.; Turek, S.: Benchmark Computations of Laminar Flow around a Cylinder. In *Flow Simulation with High-Performance Computers II.* (Hirschel, E.H., Ed.), Vieweg Verlag, Braunschweig, 1996.

[168] Shakib, F.; Hughes, T.J.R.; Johan, Z.: A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier-Stokes equations. *Comp. Methods Appl. Mech. Engrg.*, **89**, 141 – 219, 1991.

[169] Shepard, D.: A Two-Dimensional Interpolation Function for Irregularily Spaced Points. *Proceedings of the 23rd Nat. Conf. ACM*, USA, 517 – 523, 1968.

[170] Sibson, R.: A vector identity for the Dirichlet tesselation. *Math Proc Cambridge Philos Soc*, **87**, 151 – 155, 1980.

[171] Simkins, D.C.; Li, S.; Lu, H.; Liu, W.K: Reproducing kernel element method. Part IV: Globally compatible C^n $(n \geq 1)$ triangular hierarchy. *Comp. Methods Appl. Mech. Engrg.*, **193**, 1013 – 1034, 2004.

[172] Steger, J.L.; Benek, J.A.: On the Use of Composite Grid Schemes in Computational Aerodynamics. *Comp. Methods Appl. Mech. Engrg.*, **64**, 301 – 320, 1987.

[173] Strouboulis, T.; Babuška, I.; Copps, K.: The design and analysis of the Generalized Finite Element Method. *Comp. Methods Appl. Mech. Engrg.*, **181**, 43 – 69, 2000.

[174] Strouboulis, T.; Copps, K.; Babuška, I.: The Generalized Finite Element Method. *Comp. Methods Appl. Mech. Engrg.*, **190**, 4081 – 4193, 2001.

[175] Sukumar, N.; Moran, B.; Belytschko, T.: The natural element method in solid mechanics. *Internat. J. Numer. Methods Engrg.*, **43**(5), 839 – 887, 1998.

[176] Sukumar, N.; Moran, B.; Semenov, A.Y.; Belikov, V.V.: Natural neighbour Galerkin methods. *Internat. J. Numer. Methods Engrg.*, **50**, 1 – 27, 2001.

[177] Swegle, J.W.: Smoothed Particle Hydrodynamics Stability Analysis. *J. Comput. Phys.*, **116**, 123 – 134, 1995.

[178] Tezduyar, T.E.: Stabilized Finite Element Formulations for Incompressible Flow Computations. In *Advances in Applied Mechanics.* (Hutchinson, J.W.; Wu, T.Y., Eds.), Vol. 28, Academic Press, New York, NY, 1992.

[179] Tezduyar, T.E.; Mittal, S.; Ray, S.E.; Shih, R.: Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements. *Comp. Methods Appl. Mech. Engrg.*, **95**, 221 – 242, 1992.

[180] Tezduyar, T.E.; Osawa, Y.: Finite element stabilization parameters computed from element matrices and vectors. *Comp. Methods Appl. Mech. Engrg.*, **190**, 411 – 430, 2000.

[181] Tezduyar, T.E.; Park, Y.J.: Discontinuity-capturing finite element formulations for nonlinear convection-diffusion-reaction equations. *Comp. Methods Appl. Mech. Engrg.*, **59**, 307 – 325, 1986.

[182] Tiwari, S.; Kuhnert, J.: Finite pointset method based on the projection method for simulations of the incompressible Navier-Stokes equations. In *Meshfree Methods for Partial Differential Equations*. (Griebel, M.; Schweitzer, M.A., Eds.), Vol. 26, *Lectures Notes in Computational Science and Engineering*, Springer Verlag, Berlin, 2002.

[183] Turek, S.: *Efficient Solvers for Incompressible Flow Problems*, Vol. 6, *Lectures Notes in Computational Science and Engineering*. Springer Verlag, Berlin, 1999.

[184] Wagner, G.J.; Liu, W.K.: Application of essential boundary conditions in mesh-free methods: A corrected collocation method. *Internat. J. Numer. Methods Engrg.*, **47**, 1367 – 1379, 2000.

[185] Wagner, G.J.; Liu, W.K.: Hierarchical enrichment for bridging scales and mesh-free boundary conditions. *Internat. J. Numer. Methods Engrg.*, **50**, 507 – 524, 2001.

[186] Wall, W.A.; Ramm, E.: Fluid-Structure Interaction Based upon a Stabilized (ALE) Finite Element Method. *4th World Congress on Computational Mechanics – New Trends and Applications*, CIMNE, Barcelona, Spain, 1998.

[187] Wang, Z.J.; Parthasarathy, V.: A Fully Automated Chimera Methodology for Multiple Moving Body Problems. *Int. J. Numer. Methods Fluids*, **33**, 919 – 938, 2000.

[188] Wood, W.L.: *Practical Time-Stepping Schemes*. Clarendon Press, Oxford University Press, Oxford (UK), 1990.

[189] Wu, C.K.C.; Plesha, M.E.: Essential boundary condition enforcement in meshless methods: Boundary flux collocation method. *Internat. J. Numer. Methods Engrg.*, **53**, 499 – 514, 2002.

[190] Zeidler, E.: *Applied Functional Analysis—Main Principles and Their Applications*, Vol. 109, *Applied Functional Sciences*. Springer, Berlin, 1995.

[191] Zienkiewicz, O.C.; Taylor, R.L.: *The Finite Element Method*, Vol. 1 – 3. Butterworth-Heinemann, Oxford, 2000.

Curriculum Vitae

Name: Thomas-Peter Fries
(e-mail: t.fries@tu-bs.de)

Birth: 18.04.1976 in Lübeck, Germany

Schools:

1982-86	Primary school: Grundschule Utkiek in Lübeck
1986-92	Highschool: Travegymnasium Lübeck
1992-95	Highschool: Gewerbeschule III Lübeck (technical highschool)

Military service: Oct. 1995 – July 1996

University:

Oct. 1996 – Dec. 2001	Study of civil engineering at the Technical University of Braunschweig, degree: Diploma
Oct. 1999 – Dec. 2001	Parallel study of computational sciences in engineering at the Technical University of Braunschweig, degree: MSc
May 2002 – April 2005	Scholarship holder in a graduate college for fluid-structure interaction at the Technical University of Braunschweig, degree: Dr.-Ing.

In the sommer-semester 1999, I spent one semester at the University of Alicante in Spain. From April 2001 – December 2001, I studied at the Chuo University in Tokyo, Japan, where I made a Master's thesis on "Implementation of the Free Outflow Boundary Condition in a parallel, three-dimensional Navier-Stokes solver".

Conference participations:

March 2003	Annual meeting of the GAMM (German society for applied mathematics) in Padua, Italy
Sept. 2003	Workshop "Meshfree Methods for Partial Differential Equations" in Bonn, Germany
March 2004	Annual meeting of the GAMM in Dresden, Germany
June 2004	Conference "Advanced Problems in Mechanics" (APM) in St. Petersburg, Russia
Sept. 2004	World Congress on Computational Mechanics (WCCM) VI in Beijing, China
March 2005	Annual meeting of the GAMM in Luxemburg
May 2005	ECCOMAS thematic conference "Coupled Problems" in Santorini, Greece
June 2005	Third MIT-Conference on Computational Fluid and Solid Mechanics in Boston (Cambridge), USA

Braunschweig Series on Mechanics – BSM

Published reports in this series:

1–1990 Plonski, Thomas:
Dynamische Analyse von schnelldrehenden Kreiszylinderschalen

2–1991 Wegener, Konrad:
Zur Berechnung grosser plastischer Deformationen mit einem Stoffgesetz vom Überspannungstyp

3–1992 Gröhlich, Hubert:
Finite-Element-Formulierung für vereinheitlichte inelastische Werkstoffmodelle ohne explizite Fliessflächenformulierung

4–1992 Hesselbarth, Hanfried:
Simulation von Versetzungsstrukturbildung, Rekristallisation und Kriechschädigung mit dem Prinzip der zellulären Automaten

5–1992 Schlums, Hartmut:
Ein stochastisches Werkstoffmodell zur Beschreibung von Kriechen und zyklischem Verhalten metallischer Werkstoffe

6–1992 Kublik, Frithjof:
Vergleich zweier Werkstoffmodelle bei ein- und mehrachsigen Versuchsführungen im Hochtemperaturbereich

7–1992 Bechtloff, Jürgen:
Interpolationsverfahren höheren Grades für Robotersteuerungen

8–1993 Müller, Michael:
Dreidimensionale elastodynamische Analyse von Tanks mit fluidbenetzten Einbauten

9–1993 Senker, Peter:
Stabilitätsanalyse elastischer RotorsystemeStabilitätsanalyse elastischer Rotorsysteme

10–1993 Cheng, Weimin:
Schallabstrahlung einer schwingenden Reisner/Mindlin Platte

11–1993 Wiebe, Thomas:
Wellenausbreitung in poroelastischen Medien: Untersuchung mit Randintegralgleichungen

12–1993 Hahne, Matthias:
Beschreibung der plastischen Längsdehnung bei Torsion mit einem makroskopischen Stoffgesetz

13–1993 Heisig, Gerald:
Zum statischen und dynamischen Verhalten von Tiefbohrsträngen in räumlich gekrümmten Bohrlöchern

14–1994 de Araújo, Francisco Célio:
Zeitbereichslösung linearer dreidimensionaler Probleme der Elastodynamik mit einer gekoppelten BE/FE-Methode

15–1994 Kristen, Martin:
Untersuchungen zur elektrischen Ansteuerung von Formgedächtnis-Antrieben in der Handhabungstechnik

16–1994 Latz, Kersten:
Dynamische Interaktion von Flüssigkeitsbehältern und Baugrund

17–1994 Jäger, Monika:
Entwicklung eines effizienten Randelementverfahrens für bewegte Schallquellen

18–1994 August, Martin:
Schwingungen und Stabilität eines elastischen Rades, das auf einer nachgiebigen Schiene rollt

19–1995 Erbe, Matthias:
Zur Simulation von Risswachstum in dreidimensionalen, elastisch-plastischen Strukturen mit der Methode der Finiten Elemente

20–1995 Gerdes, Ralf:
Ein stochastisches Werkstoffmodell für das inelastische Materialverhalten metallischer Werkstoffe im Hoch- und Tieftemperaturbereich

21–1995 Tröndle, Georg:
Effiziente Schallberechnung mit einem adaptiven Mehrgitterverfahren für die 3-D Randelementmethode

22–1996 Degenhardt, Richard:
Nichtlineare dynamische Bauwerksprobleme und Interaktion mit dem Baugrund

23–1996 Feise, Hermann Josef:
Modellierung des mechanischen Verhaltens von Schüttgütern

24–1996 Haubrok, Dietmar:
Reibungsfreie Kontaktprobleme der 2-D Elastostatik und -dynamik als Optimierungsaufgabe mit REM-Matrizen

25–1996 Lehmann, Lutz:
Numerische Simulation der Spannungs- und Geschwindigkeitsfelder in Silos mit Einbauten

26–1996 Klein, Ralf:
Dynamische Interaktion von dünnwandigen Tragwerken und Boden mit Abschirmschlitzen

27–1996 Kopp, Thilo:
Simulation grosser inelastischer Deformationen bei Torsionsversuchen

28–1997 Harder, Jörn:
Simulation lokaler Fliessvorgänge in Polykristallen

29–1997 Lewerenz, Malte Christian:
Zur numerischen Behandlung von Werkstoffmodellen für zeitabhängig plastisches Materialverhalten

30–1997 Meywerk, Martin:
Stabilität und Verschleiss bei auf Schienen laufenden Eisenbahnradsätzen

31–1997 Plagge, Frank:
Nichtlineares, inelastisches Verhalten von Spiralseilen

32–1997 Neubert, Michael:
Richtungsregelung beim Tiefbohren

33–1998 Sangi, Daryoush:
Die Versetzungsstrukturbildung in Metallen

34–1998 Thielecke, Frank:
Parameteridentifizierung von Simulationsmodellen für das viskoplastische Verhalten von Metallen - Theorie, Numerik, Anwendung -

35–1998 Vietgen, Jürgen:
Numerische Simulation duktilen Risswachstums unter Berücksichtigung von Schädigung

36–1998 Läer, Bernard:
Einfluss transienter Anregungen auf die Zylinderkopf-Akustik

37–1998 Scheld, Christian:
Auswirkungen dynamischer Interaktionen auf das Schwingungsverhalten von Tanks

38–1999 Baaran, Jens:
Schallfeldanalyse bei sich bewegenden schallerzeugenden Körpern

39–1999 Daros, Carlos:
Wellenausbreitung in unendlich ausgedehnten piezoelektrischen Medien mit transversal isotroper Symmetrie

40–2000 Nils, Wagner:
Untersuchung der Boden-Fahrzeug-Interaktion mit gekoppelten Rand- und Finite-Element-Methoden

41–2001 Langer, Sabine:
Schalltransmission durch Isolierverglasung

42–2001 Schacht, T.:
Orientierungsabhängige Rissbildung in duktilen Metallen

43–2002 Ackerman, Lutz:
Simulation der Schalltransmission durch Wände

44–2002 Barthold, Franz-Joseph:
Zur Kontinuummechanik inverser Geometrieprobleme

45–2002 Böhrnsen, Jens-Uwe:
Dynamisches Verhalten von Schüttgütern beim Entleeren aus Silos

46–2002 Zorn, C.:
Plastisch instabile Verformung aufgrund dynamischer Reckalterung und korrelierten Versetzungsgleitens

47–2002 Küsel, M.:
Wellige Verschleissmuster auf Laufflächen von Eisenbahnrädern

48–2002 Bross, S.:
Versetzungsdynamik und Reckalterung als Ursache instabilen Materialverhaltens beim Portevin-Le Chatelier-Effekt

49–2002 Vesper, Matthias:
Modellierung von Korn und Korngrenze in polykristallinen Gefügen

50–2003 Steindorf, Jan:
Partitionierte Verfahren für Probleme der Fluid-Struktur Wechselwirkung

51–2003 Meyer, Marcus:
Reduktionsmethoden zur Simulation des aeroelastischen Verhaltens von Windkraftanlagen

52–2003 Wittich, Hauke:
Inhomogene Gleitung bei nahgeordneten CuAl-Legierungen

53–2003 Hupfer, Knut:
Einfluss der Mikrostruktur auf die Festigkeitseigenschaften von Metallschäumen

54–2004 Schmelzer, Martin:
Identifikation der Parameter von Zeitbereichsmodellen linear-viskoelastischer Werkstoffe

55–2004 Ostendorf, Michael:
Geschwindigkeitsmessungen in Silos mit der Particle Image Velocimetry

56–2004 Engelhardt, Marek:
Numerische Verfahren zur Identifizierung von Fehlstellen aus Randdaten

57–2005 Keese, Andreas:
Numerical Solution of Systems with Stochastic Uncertainties—A General Purpose Framework for Stochastic Finite Elements

58–2005 Pryl, Dobromil:
Influences of Poroelasticity on Wave Propagation: A Time Stepping Boundary Element Formulation

59–2005 Fries, Thomas-Peter:
A Stabilized and Coupled Meshfree/Meshbased Method for Fluid-Structure Interaction Problems